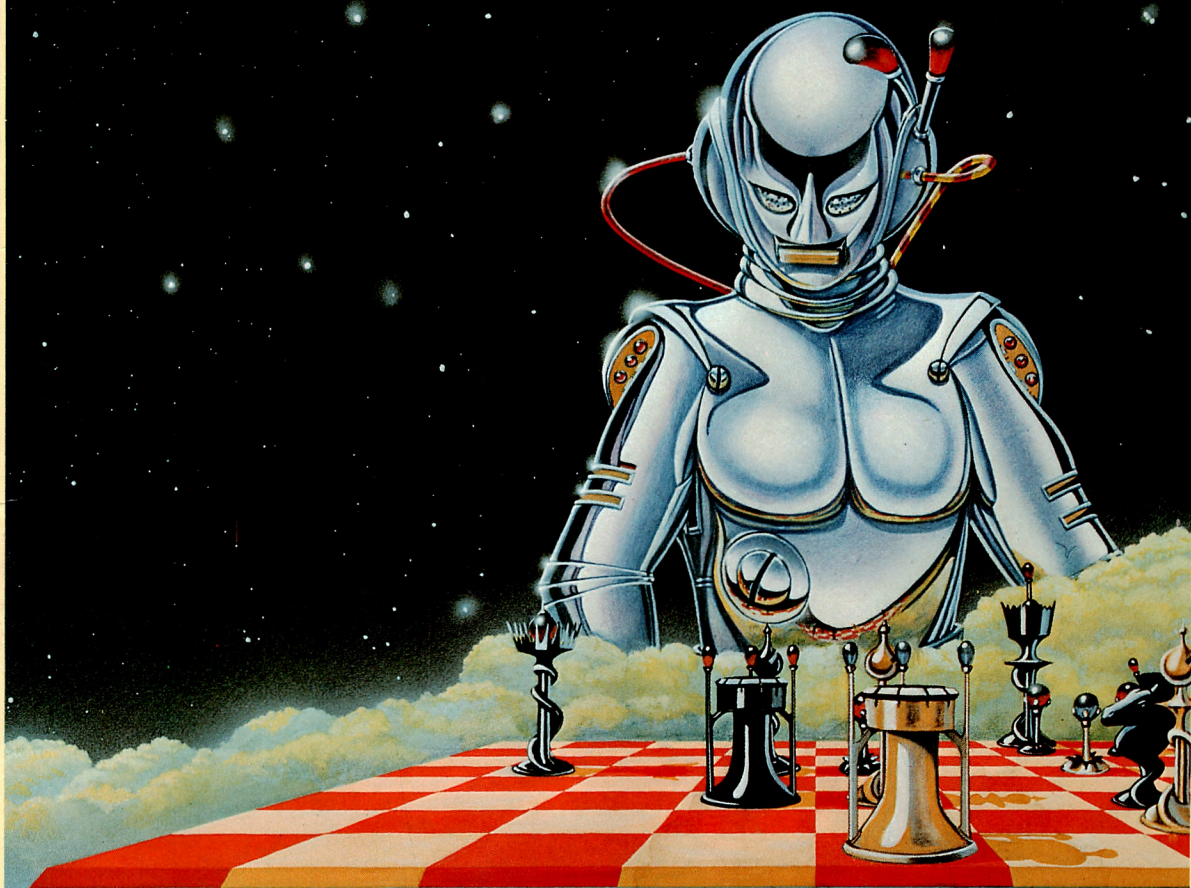


Challenging Games

Games
for the
Commodore 64

for the
Commodore 64



William A Roberts

Challenging Games

*for the
Commodore 64*



Challenging Games

*for the
Commodore 64*

William A Roberts

INTERFACE PUBLICATIONS

INTERFACE PUBLICATIONS
44-46 Earls Court Road
LONDON W8 6EJ

Distributed in Australia by
Pitman Publishing Pty Ltd
(Incorporated in Victoria)

158 Bouverie Street
Carlton
Victoria 3053

Level 11
Town Hall House
452-462 Kent Street
Sydney
New South Wales 2000

9th Floor
National Bank Building
420 George Street
Brisbane
Queensland 4000

© William A Roberts 1983

First published 1983

ISBN 0 907563 48 1

The programs in this book have been included for their instructional value. They have been tested with care but are not guaranteed for any particular purpose. While every care has been taken, the publishers cannot be held responsible for any running mistakes which may occur.

ALL RIGHTS RESERVED

No use whatsoever may be made of the contents of this volume – programs and/or text – except for private study by the purchaser of this volume, without the prior written permission of the copyright holder.

Reproduction in any form or for any purpose is forbidden.

Printed in Australia by Brown Prior Anderson Pty Ltd
Burwood Victoria

CONTENTS

INTRODUCTION	vi
RABBIT TRAP	1
PLAGUE SPOT	3
ASTRO FIGHTER	7
WARLOCK'S CASTLE	11
VALLEY OF THE SPIDERS	16
SHARKFIN	20
TUNNELS OF THAR	26
CAVERNS OF TERROR	31
REVERSI	36
COMMODORE CHECKERS	41
BLOCKOUT	47
STOCK EXCHANGE	52
EARTH BASE ONE	55
ASTEROID	58
GOLF 64 STYLE	64
ELITE GRAMOPHONE COMPANY	67

INTRODUCTION

This book was written with a sense of excitement, a sense of excitement which will transmit itself to you the moment you get the first program up and running.

From Caverns of Terror and Tunnels of Thar to Warlock's Castle, you'll find here a satisfying collection of challenging games for your Commodore 64.

Whether you're looking for shoot-em-ups, alien-blasters, outwit-the-machines or other games, you'll find them here, just itching to get off the printed page and into your micro.

It's about time you caught the sense of excitement, and gave way to the urging of the games.

Time to start playing, and start enjoying, some great games on your Commodore 64.

Tim Hartnell,

Melbourne, 1983

RABBIT TRAP

In this game the computer will try to trap you, a frightened rabbit, by placing obstacles in your path, finally boxing you in so that you can't move. The object of the game is to keep running for as long as possible. Your score at the end of the game is related to how long you remained free.

To move your rabbit use the "A" key to move up and the "Z" key to move down the screen. The comma and the full stop keys move the rabbit left and right respectively.

Lines 1000 to 1050 GET your input from the keyboard and increment your position on the screen. Lines 1060 to 1090 check your position each turn to see if you are surrounded by obstacles, if F equals one you are trapped and the program goes to line 2000. You are then told your final score.

Line 1100 increments your score each time the program goes through the loop which moves you about the screen. The location of the obstacles is picked in line 1110. The second statement in this line prevents an obstacle from being POKEd into a spot already occupied or into one of the walls around the playing area. Line 1130 POKEs the obstacle onto the screen.

```
10 REM RABBIT TRAP
20 GOSUB 3000
30 POKE 53280,6:POKE 53281,8
1000 GET A$:IF A$="" THEN A$=B$
1010 E=C:Q=S
1020 IF A$="," THEN S=S+1:C=C+1
1030 IF A$="," THEN S=S-1:C=C-1
1040 IF A$="A" THEN S=S-40:C=C-40
1050 IF A$="Z" THEN S=S+40:C=C+40
1060 W=PEEK(S):F=0
1070 IF W=102 THEN C=E:S=Q
1075 IF PEEK(Q+40)=102 THEN F=F+.25
1076 IF PEEK(Q-40)=102 THEN F=F+.25
1077 IF PEEK(Q-1)=102 THEN F=F+.25
1078 IF PEEK(Q+1)=102 THEN F=F+.25
```

```

1090 IF F=1 THEN 2000
1100 SC=SC+1
1110 H=1024+INT(570*RND(1)):IF PEEK(H)=102
    THEN 1110
1120 J=H+54272
1130 POKE H,102:POKE J,0
1200 POKE E,1
1210 POKE C,2
1215 POKE S,88
1220 B$=A$
1230 GOTO 1000
2000 FOR DD=1 TO 2000:NEXT DD
2010 PRINT "□":REM CLEAR SCREEN
2020 POKE 53280,5:POKE 53281,2
2030 PRINT "XXXXXXXX" TAB(15) "XCAUGHT YOU":
    REM ORANGE
2040 PRINT "XX" TAB(11) "XYOU SCORED";SC*236:
    REM LT BLUE
2050 GOTO 2050
3000 PRINT "□":REM CLEAR SCREEN
3010 POKE 53280,0:POKE 53281,4
3020 PRINT "XXXXXXXX" TAB(7) "X##### RABBIT TRAP
    #####":REM BLACK
3030 FOR DD=1 TO 2000:NEXT
3035 PRINT "□":REM CLEAR SCREEN
3040 S=1110:C=S+54272
3050 B$="."
3060 FOR GG=1024 TO 1063:POKE GG,102:POKE GG+54272,
    1:NEXT
3070 FOR GG=1584 TO 1623:POKE GG,102:POKE GG+54272,
    1:NEXT
3080 FOR GG=1064 TO 1544 STEP 40:POKE GG,102:POKE
    GG+54272,1:NEXT
3090 FOR GG=1103 TO 1583 STEP 40:POKE GG,102:POKE
    GG+54272,1:NEXT
3100 RETURN

```

PLAGUE SPOT

This game combines moving graphics and a random number generator to present you with an interesting challenge. The program begins by drawing a border around the screen (lines 40 to 80) then places 125 black or white plague spots on the screen. This is done by the routine in lines 120 to 150. The FOR/NEXT loop cycles through the random number generator in line 130. This number is used for the screen location of a plague spot. The random number generator in line 140 determines whether the spot will be white or black.

You then appear on the screen near the top left hand corner, you are represented by a purple "Clubs" symbol. You have to run from your starting position at the top of the screen to the gate at the bottom right hand corner of the screen. Along the way you have to avoid running into plague spots or the wall around the screen. Colliding with the wall or a plague spot will result in ten points being added to your score. The aim of the game is to cross the screen with as low a score as possible. To make your task even more difficult lines 230 and 240 continue to place plague spots on the screen during the running of the game.

You control your little figure on the screen with four keys on the keyboard. On the left of the keyboard, the "A" key moves your man up the screen and the "Z" key moves him down. On the right of the keyboard, the keys with the greater than and less than symbols are used for moving right and left respectively.

Line 110 sets the starting position of your little man. Lines 170 to 210 accept your input from the keyboard, make the necessary adjustment to your position and then use PEEK to see if you have run into a plague spot or a wall.

Lines 215 and 216 stop you from accidentally running off the screen. The PEEK in line 219 checks to see whether you have hit the exit gate at the bottom of the screen.

The routines from line 1000 to line 1510 are called up after you run into a wall or a plague spot. They call up the sound routine, prevent your man from moving onto the plague spot or through the wall and also increment your score.


```

5 REM FLAGUE SPOT
10 GOSUB 6500
15 HS=999999
20 PRINT "J" REM CLEAR SCREEN
25 POKE 53280,0:POKE 53281,6
30 SC=0
35 B$="."
40 FOR PP=1024 TO 1063:POKE PP,81:POKE PP+54272,8:
  NEXT PP
50 FOR PP=1064 TO 1944 STEP 40:POKE PP,81:POKE
  PP+54272,8:NEXT PP
60 FOR PP=1103 TO 1983 STEP 40:POKE PP,81:POKE
  PP+54272,8:NEXT PP
70 FOR PP=1984 TO 2020:POKE PP,81:POKE PP+54272,8:
  NEXT PP
80 FOR PP=2021 TO 2023:POKE PP,121:POKE PP+54272,1
  :NEXT PP
110 SL=1066
120 FOR T=1 TO 125
130 Z=1070+INT(910*RND(1))
140 POKE Z,81:POKE Z+54272,INT(2*RND(1))
150 NEXT T
160 M=SL
170 GET A$
175 IF A$<>"A" AND A$<>"Z" AND A$<>"," AND A$<>". "
  THEN A$=B$
175 B$=A$
180 IF A$="." THEN SL=SL+1:IF PEEK (SL)=81 THEN
  GOSUB 1000
190 IF A$="," THEN SL=SL-1:IF PEEK (SL)=81 THEN
  GOSUB 1100
200 IF A$="Z" THEN SL=SL+40:IF PEEK (SL)=81 THEN
  GOSUB 1200
210 IF A$="A" THEN SL=SL-40:IF PEEK (SL)=81 THEN
  GOSUB 1300
215 IF SL<1024 THEN SL=1024
216 IF SL>2023 THEN SL=1024

```

```

219 IF PEEK(SL)=121 THEN GOTO 5000
220 POKE N,32:POKE SL,98:POKE SL+54272,4
230 J=INT(910*RND(1))
240 POKE 1066+J,81:POKE 1066+54272+J,INT(2*RND(1))
500 GOTO 100
1000 GOSUB 6000
1010 SL=SL-1
1020 SC=SC+1
1030 GOTO 1500
1100 GOSUB 6000
1110 SL=SL+1
1120 SC=SC+1
1130 GOTO 1500
1200 GOSUB 6000
1210 SL=SL-40
1220 SC=SC+1
1230 GOTO 1500
1300 GOSUB 6000
1310 SL=SL+40
1320 SC=SC+1
1500 PRINT "■=SCORE: ";10*SC:REM HOME WHITE
1505 IF SC>99 THEN 5000
1510 RETURN
5000 FOR DD=1 TO 2000:NEXT DD
5010 PRINT "□":REM CLEAR SCREEN
5020 POKE 53280,6:POKE 53281,4
5030 PRINT "■" :REM CURSOR DOWN
5040 FOR PP=1 TO 80:PRINT "■" CHR$(166):NEXT PP:
    REM RED
5050 IF SC<HS THEN HS=SC
5060 PRINT "■" TAB(10) "YOUR SCORE WAS ";10*SC:
    REM LT GREEN
5070 PRINT
5080 PRINT TAB(8) "■" THEN BEST SO FAR IS ";10*HS:
    REM YELLOW
5090 PRINT
5095 FOR PP=1 TO 80:PRINT "■" CHR$(166):NEXT PP:
    REM RED
5100 FOR DD=1 TO 3000:NEXT DD
5110 GOTO 20
6000 POKE 54296,15

```

```

6010 POKE 54277,17:POKE 54278,17
6030 POKE 54276,17
6040 POKE 54273,38:POKE 54272,126
6050 FOR DD=1 TO 15:NEXT DD
6060 POKE 54276,0:POKE 54277,0:POKE 54278,0
6070 RETURN
6500 PRINT "D":REM CLEAR SCREEN
6510 POKE 53280,1:POKE 53281,0
6520 PRINT "XXXXXXXXXX":REM CURSOR DOWN
6530 PRINT TAB(14) "PLAGUE SPOT":REM REV ON
6540 POKE 54296,15
6550 W=54276:A=54277:S=54278
6560 HF=54273:LF=54272
6570 GOSUB 7000
6580 POKE HF,3:POKE LF,155
6590 FOR DD=1 TO 750:NEXT DD
6600 POKE W,0:POKE A,0:POKE S,0
6610 FOR DD=1 TO 75:NEXT DD
6620 GOSUB 7000
6630 POKE HF,4:POKE LF,208
6635 FOR DD=1 TO 500:NEXT DD
6640 POKE W,0:POKE A,0:POKE S,0
6650 FOR DD=1 TO 75:NEXT DD
6660 GOSUB 7000
6670 POKE HF,3:POKE LF,54
6680 FOR DD=1 TO 1200:NEXT DD
6690 POKE W,0:POKE A,0:POKE S,0
6700 RETURN
7000 POKE A,130:POKE S,130:POKE W,33
7010 RETURN

```

ASTRO FIGHTER

We journey into space now so that you can become the pilot of a V-wing Astro fighter, patrolling a sector of space around your home planet. Nasty yellow alien numbers appear in various places around you. They only remain visible for a short, and varying length of time before becoming invisible. Your task is to catch as many of these numbers as possible, by flying your fighter over them, before the time limit runs out.

The up and down movement of your craft is controlled by the “A” and “Z” keys respectively. The left and right movements are controlled by the comma and full stop keys. These keys were chosen because the symbols above the comma and full stop show the direction of movement.

Your score at the end of the game relates to the values of the numbers you have caught.

Lines 120 to 150 are used to check your keyboard input and then increment your position on the screen. The statement at the end of each of these lines changes the appearance of your fighter as it alters direction. When you are flying toward the top of the screen, line 140 changes the symbol used to depict your fighter in line 220 to the letter A. When you are flying toward the bottom of the screen a letter V is used. The left and right movement uses the greater than and less than symbols to represent your fighter.

Lines 160 and 170 check to see whether you have hit one of the walls around the screen. If you have, your screen position (variable S) is not altered until you change direction. Line 180 checks to see whether you have run over a number. If you have it multiplies the code of the number by eighteen and adds it to your score.

The routine from line 1000 to 1200 sets up the start of the game. Line 1010 sets the screen starting position, line 1020 sets the colour memory starting position. Line 1030 gives the starting direction of your fighter by setting B\$ equal to the full stop key. This has the effect of continuous movement. When the program reaches line 100 if there is no input from the keyboard the value of B\$ is used to set up A\$. This value of A\$ is then used as the input. Line 230 then sets the current value of A\$ equal to B\$. Next time through the loop, line 100 keeps you moving in the same direction if you do not input a new direction.

Lines 1040 to 1070 draw the border around the screen. Line 1080 diverts the program around the DIM statements in the following line to prevent the arrays from being reDIMmed on subsequent running of the program. ReDIMming an array causes a systems error in your computer. This is avoided by using the high score variable HS. The first time the game is run HS equals zero so the arrays are set up. The next time you play the game HS holds the value of your score for the first game. This then meets the condition in line 1080 and the program skips around the DIM statements.

Lines 1090 to 1160 set up the three arrays. The A array contains 55 numbers chosen at random between one and nine. These numbers become the aliens. The B array contains 55 screen locations chosen at random and the C array holds the corresponding colour memory locations. Line 1150 prevents the aliens from being positioned in a wall.

Variable CC in line 1170 is used as the counter to run through the arrays, POKEing the aliens onto the screen. CC is incremented in line 190 and tested in line 200. When CC equals 55 the arrays are empty and the game ends.

```

10 REM ASTRO FIGHTER
15 HS=0
20 GOSUB 2000
50 POKE 53280,2:POKE 53281,6
60 GOSUB 1000
100 GET A$:IF A$="" THEN A$=B$
110 PRINT "H*SCORE:":SC:REM HOME WHITE
115 E=C:Q=S:V1=V
120 IF A$="." THEN S=S+1:C=C+1:V=62
130 IF A$="," THEN S=S-1:C=C-1:V=60
140 IF A$="A" THEN S=S-40:C=C-40:V=1
150 IF A$="Z" THEN S=S+40:C=C+40:V=22
160 W=PEEK(S)
170 IF W=102 THEN C=E:S=Q:V=V1
175 F=0
180 IF W>48 AND W<58 THEN SC=SC+W*18
185 JJ=RND(1)
190 IF JJ>.92 THEN POKEC(CC),0:POKEB(CC),48:CC=CC+1
    :POKEB(CC),A(CC):POKEC(CC),7

```



```

200 IF CC=55 THEN 300
205 POKE S,32
210 POKE E,0
220 POKE C,7:POKE S,V
230 B#=A#
240 GOTO 100
300 FOR DD=1 TO 1000:NEXT DD
310 PRINT "J":REM CLEAR SCREEN
320 POKE 53280,2:POKE 53281,5
330 PRINT "XXXXX" TAB(14) "TIME IS UP":REM REV
    ON BLACK
340 PRINT "MM" TAB(12) "YOU SCORED":SC
345 IF HS<SC THEN HS=SC
350 PRINT "M" TAB(12) "HIGH SCORE IS:":HS
360 FOR DD=1 TO 2500:NEXT DD
370 GOTO 50
1000 PRINT "J":REM CLEAR SCREEN
1010 S=1100
1020 C=55380
1030 B#="."
1040 FOR PP=1024 TO 1063:POKE PP,102:POKE PP+54272,
    1:NEXT PP
1050 FOR PP=1064 TO 1944 STEP 40:POKE PP,102:POKE
    PP+54272,1:NEXT PP
1060 FOR PP=1103 TO 1983 STEP 40:POKE PP,102:POKE
    PP+54272,1:NEXT PP
1070 FOR PP=1984 TO 2023:POKE PP,102:POKE PP+54272,
    1:NEXT PP
1080 IF HS<>0 THEN GOTO 1100
1090 DIM A(55),B(55),C(55)
1100 FOR Z=1 TO 55
1120 A(Z)=INT(9*RND(1))+49
1130 B(Z)=1065+INT(910*RND(1))
1140 C(Z)=B(Z)+54272
1150 IF PEEK(B(Z))=102 THEN 1130
1160 NEXT Z
1170 CC=1
1180 V=60
1190 SC=0:IF HS=0 THEN HS=1

```

```
1200 RETURN
2000 PRINT "C":REM CLEAR SCREEN
2010 POKE 53280,1:POKE 53281,4
2020 PRINT "MMMMMMMM":REM DOWN CURSOR
2030 PRINT TAB(8) "##### ASTRO FIGHTER #####":
    REM BLUE
2040 FOR DD=1 TO 1500:NEXT DD
2050 RETURN
```

WARLOCK'S CASTLE

At the beginning of this game you find yourself standing on the doorstep of a castle occupied by an evil warlock. The warlock has cast a spell over the surrounding countryside and, being a gallant hero, your task is to enter the castle and track down the warlock. The only way to break the spell and to free the country is to kill the warlock with a magic arrow fired from your trusty bow. The warlock is very powerful and has surrounded himself with a bodyguard of trolls, so the task will not be easy to complete.

Being a magic castle, it has no doors. The magic doorstep will transport you inside the castle and deposit you in a room chosen at random. You will then be told which rooms you can get to from your present position. By shouting out a room number at the top of your voice (or by entering the number into your computer), you will be carried off to that room. You will also be warned if the warlock or his bodyguard are nearby.

If you enter the room the warlock is in you will suffer a fate worse than death. To kill the warlock you have to fire a magic arrow through a wall from one room to another. To do this put a minus sign in front of the room number when you enter it. For instance -32. You start the game with three arrows so use them wisely. However you may be lucky enough to find an arrow or two as you wander around the castle.

The computer will tell you when you are getting close to the warlock and his trolls. It will also tell you which rooms you can enter. Some rooms contain magic spells. These rooms will immediately send you off somewhere else.

```
10 REM WARLOCK'S CASTLE
15 REM "X" IS DOWN CURSOR
20 GOSUB 960
30 PRINT "J":REM CLEAR SCREEN
40 DIM A(30)
50 DIM B(30)
60 G=3
70 FOR Z=0 TO 30
80 A(Z)=0
90 NEXT Z
```

```

100 FOR Z=0 TO 10
110 A=INT(30*RND(1))+1
120 IF A(A)=1 THEN 110
130 A(A)=1
140 B(Z)=A+11
150 NEXT Z
155 IF B(10)=0 THEN B(10)=INT(25*RND(1))+1
160 PRINT "XOXOX" TAB(10) "THERE IS MAGIC HERE":
    REM BLUE
170 FOR DD=1 TO 1000:NEXT DD
180 X=(INT(16*RND(1))+1)*2+10
190 PRINT "X" TAB(5) "YOU ARE BEING SENT TO ROOM":
    X:REM CURSOR DOWN
200 FOR DD=1 TO 1000:NEXT DD
210 Y=7
220 PRINT "J":REM CLEAR SCREEN
230 PRINT "X" TAB(8) "YOU ARE NOW IN ROOM #":X:
    REM BLUE
240 A=-1
250 FOR Z=0 TO 15
260 IF B(Z)=X THEN A=INT(Z/5)
270 NEXT Z
280 IF A=0 THEN 160
295 IF A<>1 THEN GOTO 350
300 FOR DD=1 TO 1000:NEXT DD
310 PRINT "XOXOX" TAB(9) "THERE ARE TROLLS HERE":
    REM BLACK
320 FOR DD=1 TO 1000:NEXT DD
330 PRINT "XOXOX" TAB(9) "THEY HAVE KILLED YOU"
340 GOTO 770
350 IF A=2 AND RND(1)>.4 THEN PRINT "XOXOXYOU HAVE
    FOUND THE WARLOCK":GOTO 690
375 F=X-1:Q=X+1:H=X+Y
376 IF F<1 THEN F=1
377 IF Q<1 THEN Q=1
378 IF H<1 THEN H=1
380 PRINT "X" TAB(4) "YOU CAN MOVE TO: #":F;"#":Q:
    "#":H
390 FOR Z=0 TO 2
400 A(Z)=0

```

```

410 NEXT Z
420 FOR Z=0 TO 10
430 D=B(Z)-X
440 IF ABS(D)=1 OR D=Y THEN A(INT(Z/5))=1
450 NEXT Z
460 D=ABS(D)
470 IF D=2 OR D=6 OR D=8 THEN A(2)=1
480 IF A(0)=1 THEN PRINT "魔" TAB(5) "*** THERE IS
    MAGIC NEARBY ***":REM YELLOW
490 IF A(1)=1 THEN PRINT "魔" TAB(4) "*** THERE
    ARE TROLLS NEARBY ***":
495 REM BLACK
500 IF A(2)=1 THEN PRINT "魔" TAB(5) "*** THE
    WARLOCK IS NEARBY ***":REM GRAY 1
510 A=2:IF RND(1)>.8 THEN PRINT "魔" TAB(8)
    "YOU HAVE FOUND AN ARROW":G=G+1
515 REM RED
520 PRINT "魔" TAB(3) "WHICH ROOM DO YOU WISH TO
    ENTER":REM BLUE
530 INPUT M
540 PRINT "□":REM CLEAR SCREEN
550 FOR DD=1 TO 1000:NEXT DD
560 IF M<0 THEN 610
570 IF M<1 OR M>45 THEN PRINT "魔" TAB(10)
    "CANNOT MOVE THERE":GOTO 530
580 X=M
590 Y=-Y
600 GOTO 220
610 IF RND(1)>.6 THEN 630
620 IF M=-B(10) THEN 860
630 G=G-1
640 PRINT "魔魔魔" TAB(14) "YOU MISSED":REM WHITE
650 PRINT "魔" TAB(9) "YOU HAVE":G/"ARROWS LEFT"
660 FOR DD=1 TO 1000:NEXT DD
670 PRINT "□":REM CLEAR SCREEN
680 GOTO 230
690 FOR DD=1 TO 1500:NEXT:PRINT "□":REM CLEAR
    SCREEN
700 X=INT(4*RND(1))+1

```



```

710 IF X=1 THEN Y$="TOAD"
720 IF X=2 THEN Y$="TREE"
730 IF X=3 THEN Y$="RABBIT"
740 IF X=4 THEN Y$="WORM"
745 FOR ID=1 TO 1000:NEXT ID
750 PRINT "■■■■" TAB(3) "YOU HAVE BEEN TURNED INTO
      A ";Y$
770 GOSUB 2000
780 GOTO 780
860 IF A=2 AND RND(1)>.6 THEN G=G-1:GOTO 640
870 PRINT "□":REM CLEAR SCREEN
880 FOR ID=1 TO 1000:NEXT ID
890 PRINT TAB(6) "■■■■■■YOU HAVE KILLED THE WIZARD"
      :REM BLACK
900 GOSUB 3000
905 GOTO 905
910 FOR ID=1 TO 1000:NEXT ID
960 PRINT "□":REM CLEAR SCREEN
970 POKE 53280,2:POKE 53281,10
980 PRINT "■■■■■■":REM DOWN CURSOR
990 FOR X=1 TO 40:PRINT "■" CHR$(113);:NEXT:REM
      YELLOW
1000 PRINT "■■■" TAB(8) "### WARLOCK'S CASTLE ###"
      :REM BLACK
1010 PRINT "■■■"
1020 FOR X=1 TO 40:PRINT "■" CHR$(113);:NEXT:REM
      YELLOW
1030 GOSUB 4000
1040 RETURN
2000 POKE 54296,15
2010 POKE 54277,17:POKE 54278,130
2020 POKE 54276,33
2030 POKE 54273,4:POKE 54272,73
2040 FOR ID=1 TO 1000:NEXT
2050 POKE 54273,5:POKE 54272,185
2060 FOR ID=1 TO 750:NEXT
2070 POKE 54273,3:POKE 54272,155
2080 FOR ID=1 TO 1500:NEXT
2090 POKE 54276,0:POKE 54277,0:POKE 54278,0
2100 RETURN

```

```

3000 POKE 54296,15:POKE 54277,130:POKE 54278,130
3010 POKE 54276,17
3020 X=2:Y=100:XX=1:YY=10
3030 POKE 54273,X:POKE 54272,Y
3040 FOR DD=1 TO 175:NEXT
3045 T=T+1
3050 X=X+XX:Y=Y+YY
3060 IF T=9 THEN YY=-YY:XX=-XX
3065 IF T>16 THEN POKE 54276,0:POKE 54277,0:POKE
      54278,0:END
3070 GOTO 3030
4000 POKE 54296,15
4010 POKE 54277,20:POKE 54278,130
4020 POKE 54276,33
4040 POKE 54273,3:POKE 54272,54
4045 FOR DD=1 TO 500:NEXT
4050 POKE 54273,3:POKE 54272,155
4055 FOR DD=1 TO 500:NEXT
4060 POKE 54273,2:POKE 54272,220
4070 FOR DD=1 TO 750:NEXT
4080 POKE 54273,3:POKE 54272,54
4085 FOR DD=1 TO 1000:NEXT
4090 POKE 54277,0:POKE 54278,0:POKE 54276,0
4100 RETURN

```

VALLEY OF THE SPIDERS

Your aircraft has crashed in an isolated valley high in the Andes mountains. The only inhabitants of the valley are large blue spiders. These spiders are very hungry and, fortunately for you, not very bright. As they pursue you around the valley they often blunder into the numerous swamps in their hurry to catch you. Your only hope for survival is to avoid the first rush of spiders and then lure the ones chasing you into the swamps.

The screen display will show the valley boundaries, which neither you nor the spiders can cross. You will be displayed as a black CLUBS symbol and the swamps are green circles. You will recognise the spiders when you see them.

The game starts off with you running to the right across the screen. Your little man moves continually during the game and you use keys on the computer keyboard to change his direction. The "A" key moves him up the screen, the "Z" key down the screen. Press the "," key to move left and "." to move right. These keys may feel a bit odd at first but after a little while you will find that they are very easy to use.

The routine from lines 2000 to 2500 sets up the program, line 2002 changes the screen and border colours and lines 2010 to 2040 draw the top and bottom borders of the valley. Lines 2050 to 2070 select and place the spiders and swamps and also draw the sides of the valleys. Variable Z represents the swamps and Z1 is the spiders.

Lines 30 to 80 check the keyboard for your input and make the relevant direction changes.

Lines 90 to 140 use PEEK to see if you have walked into the edge of the valley or a swamp, in which case they prevent you from moving any further in that direction. These lines also check to see whether you have bumped into a spider.

Lines 160 to 290 control the movement of the spiders. These lines also use PEEK to see whether the spiders have walked into a wall, a swamp or caught up to you.

Lines 1500 to 1630 provide the end routines which tell you how well or how badly you performed.

Valley of the Spiders is a fun game and one well worth experimenting with. No computer game program is ever really finished. So please use this game as a starting point and as your programming skills grow, use it as an exercise in moving graphics. Try to add as many additional features as you can. For instance you could make the swamps fatal to our little pilot as well as the spiders. Try to add a high score feature or more hazards for the player to avoid. The possibilities are endless.

```

10 REM VALLEY OF THE SPIDERS
20 GOSUB 2000
30 GET S$:IF S$="" THEN S$=T$
40 AE=A:BE=B
50 IF S$="A" THEN A=A-40:B=B-40
60 IF S$="Z" THEN A=A+40:B=B+40
70 IF S$="." THEN A=A+1:B=B+1
80 IF S$="," THEN A=A-1:B=B-1
90 W=PEEK(A)
100 IF W=127 THEN A=AE:B=BE
110 IF W=42 THEN A=AE:B=BE
120 IF W=81 THEN A=AE:B=BE
130 POKE AE,43:POKE BE,1
140 POKE A,88:POKE B,0
150 T$=S$
160 REM SPIDER CONTROL
170 FOR R=1 TO Y
180 U=0
185 IF PEEK(Q(R))=81 THEN 260
190 IF Q(R)>A THEN U=-1:IF RND(1)>.5 THEN U=-40
200 IF Q(R)<A THEN U=1:IF RND(1)>.5 THEN U=40
210 POKE Q(R),46:POKE P(R),1
220 Q(R)=Q(R)+U:P(R)=P(R)+U
225 IF PEEK(Q(R))=88 THEN 400
230 IF PEEK(Q(R))=81 THEN TC=TC+1:GOTO 260
235 IF PEEK(Q(R))=127 THEN Q(R)=Q(R)-U:P(R)=P(R)-U
240 POKE Q(R),42:POKE P(R),14
260 IF TC=Y THEN 310
290 NEXT
294 PRINT "X":REM HOME CURSOR
295 PRINT TAB(10)"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXINITIALLY";TC;
  "OUT OF";Y
296 REM WHITE DOWN CURSOR
297 GOSUB 3000

```

```

299 SC=SC+1
300 GOTO 30
310 GOTO 1500
350 GOTO 410
400 POKE Q(R),42:POKE P(R),14
405 GOTO 1600
410 FOR DD=1 TO 3000:NEXT DD
420 RUN
1500 PRINT "C":REM CLEAR SCREEN
1510 SC=SC*100
1520 PRINT TAB(10) "XXXXXXXXXXYOU KILLED";Y;"SPIDERS"
1525 REM WHITE CURSOR DOWN
1530 PRINT TAB(11) "XYOU HAVE SURVIVED"
1540 PRINT TAB(11) "XYOUR SCORE IS";SC
1560 GOTO 410
1600 POKE 53280,11:POKE 53281,0
1605 PRINT "C":REM CLEAR SCREEN
1610 PRINT TAB(6) "XXXXXXXXXXYOU KILLED";TC;
      "SPIDERS OUT OF";Y
1620 PRINT TAB(10) "XYOU DIDN'T SURVIVE"
1630 GOTO 410
2000 REM INITIALISE
2002 POKE 53280,6:POKE 53281,11
2005 PRINT "C":REM CLEAR SCREEN
2010 FOR J=1 TO 35
2020 POKE 55295+J,4:POKE 1023+J,127
2030 POKE 56055+J,4:POKE 1783+J,127
2040 NEXT
2050 FOR J=1024 TO 1784 STEP 40:POKE J,127:POKE
      54272+J,4
2054 Z=INT(30*RND(1))+3:Z1=INT(30*RND(1))+2
2055 IF J>1024 AND J<1784 THEN POKE J+Z,81:POKE
      J+Z+54272,3
2056 T=0:IF J>1024 AND J<1784 AND RND(1)>.6 THEN
      T=1
2057 IF T=1 THEN Y=Y+1:Q(Y)=J+Z1:P(Y)=J+Z1+54272:
      POKE Q(Y),42:POKE P(Y),14
2060 POKE J+35,127:POKE 54307+J,4
2070 NEXT
2080 A=1065:B=55337:REM START POSITION OF PLAYER
2090 POKE A,88:POKE B,0

```



```
2100 T$="."
2110 SC=0
2500 RETURN
3000 POKE 54296,15
3010 POKE 54277,17
3020 POKE 54276,129
3030 POKE 54273,34:POKE 54272,75
3035 FOR T=1 TO 10:NEXT T
3040 POKE 54276,0
3050 POKE 54277,0
3060 RETURN
```

SHARKFIN

You are the Captain of the nuclear submarine USS SHARKFIN on patrol in the North Atlantic. Somewhere in the waters around you is an enemy submarine. You must find it and put it out of action with your torpedoes. To score a hit you must get as close as you can but don't get too close, a collision with the enemy submarine will be fatal to you both.

Moving your submarine and firing your torpedoes uses up your fuel supply, so play your moves carefully.

Being hit by an enemy torpedo causes a loss of damage points (DP). When your damage points reach zero (line 605) your submarine has been destroyed.

Your air supply also has a limit so that you can only stay submerged for a certain amount of time.

The computer will keep you informed of your position in the icy waters and will also give you the approximate position of the enemy submarine.

Line 870 sets the amount of fuel you have available, line 875 gives you a random number of damage points so that your ability to withstand direct hits will be different each game. Line 890 sets the time limit for the game.

Lines 900 to 920 choose the starting position of the enemy submarine and lines 930 to 950 choose your starting position. Line 750 checks for a collision and line 280 sees whether the enemy submarine is in range when you fire a torpedo.

Lines 100 to 170 accept your move and adjust your position in the ocean. Lines 190 to 240 move the enemy submarine.

Lines 670 to 850 provide the feedback after each move. Lines 760 to 840 compare your position with that of the enemy and provide the approximate direction.

```

10 REM USS SHARKFIN
20 GOSUB 1000
30 PRINT "J":REM CLEAR SCREEN
35 POKE 53280,2:POKE 53281,6
40 GOSUB 870
50 GOSUB 670
60 IF L<0 THEN GOTO 430
70 PRINT:PRINT "M" TAB(2) "WHAT IS YOUR ORDER, SIR"
80 L=L-1
90 PRINT "M" TAB(2) "N:S:E:W:T(ORPEDO):F(ORWARD):
    B(ACK)";
100 INPUT Z$
110 IF Z$="T" THEN GOSUB 280
120 IF Z$="N" THEN X=X-1
130 IF Z$="S" THEN X=X+1
140 IF Z$="E" THEN Y=Y+1
150 IF Z$="W" THEN Y=Y-1
160 IF Z$="F" THEN Z=Z-1
170 IF Z$="B" THEN Z=Z+1
180 GOSUB 510
190 A=A+INT(3*RND(1))-INT(3*RND(1))
200 IF A<1 OR A>10 THEN GOTO 190
210 B=B+INT(3*RND(1))-INT(3*RND(1))
220 IF B<1 OR B>10 THEN GOTO 210
230 C=C+INT(3*RND(1))-INT(3*RND(1))
240 IF C<1 OR C>10 THEN GOTO 230
260 GOTO 50
270 L=L-2
280 IF ABS(A-X)>3 OR ABS(B-Y)>3 OR ABS(C-Z)>3
    THEN GOSUB 970
290 FOR DD=1 TO 1500:NEXT DD
310 IF ABS(A-X)>3 OR ABS(B-Y)>3 OR ABS(C-Z)>3 THEN
    RETURN
320 PRINT "M" TAB(4) "ORDER TO FIRE UNDERSTOOD,
    SIR":REM WHITE
330 FOR DD=1 TO 1000:NEXT DD
340 PRINT "M" TAB(10) "TORPEDO RUNNING"
350 IF RND(1)>.7 THEN GOTO 400
355 PRINT "M" TAB(5):FOR TT=1 TO 6
360 PRINT ">";

```

22

```

605 IF DP<0 THEN GOTO 3500
620 RETURN
630 PRINT "M" TAB(6) "THE ENEMY MISSED US, SIR":
    REM BLACK
640 FOR DD=1 TO 2000:NEXT DD
660 RETURN
670 PRINT "J":REM CLEAR SCREEN
680 PRINT "M" TAB(6) "FUEL REMAINING:";L:REM
    YELLOW

690 TY=TY-1
700 IF TY=0 THEN GOTO 430
710 PRINT "M" TAB(6) "TIME:";TY
720 IF LC3 THEN PRINT "M" TAB(4) "FUEL IS RUNNING
    LOW"
740 PRINT "M" TAB(6) "YOU ARE AT";X;" ";Y;" ";Z
750 IF A=X AND B=Y AND C=Z THEN GOTO 480
760 PRINT TAB(6) "THE ENEMY IS ";
770 IF A<X OR B<Y THEN PRINT TAB(6) "TO THE "
775 PRINT
778 PRINT TAB(6)
780 IF A<X THEN PRINT "NORTH";
790 IF A>X THEN PRINT "SOUTH";
800 IF B<Y THEN PRINT "EAST";
810 IF B>Y THEN PRINT "WEST";
830 IF C>Z THEN PRINT " BEHIND YOU"
840 IF C<Z THEN PRINT " IN FRONT OF YOU"
850 RETURN
870 L=15+INT(20*RND(1))+1
875 DP=25+INT(30*RND(1))+1
880 T=0
890 TY=35
900 A=INT(10*RND(1))+1
910 B=INT(10*RND(1))+1
920 C=INT(10*RND(1))+1
930 X=INT(10*RND(1))+1
940 Y=INT(10*RND(1))+1
950 Z=INT(10*RND(1))+1
960 RETURN
970 FOR DD=1 TO 1000:NEXT DD
980 PRINT "M" TAB(10) "### OUT OF RANGE ###":REM
    LT RED

```

```

990 RETURN
1000 POKE 53280,6:POKE 53281,14
1005 PRINT "□":REM CLEAR SCREEN
1010 PRINT "XXXXXXXXXXXX":REM DOWN CURSOR
1020 PRINT TAB(10) "#### USS SHARKFIN ####":REM
    BLACK YELLOW BLACK

1030 POKE 54296,15:POKE 54277,17:POKE 54278,17
1040 POKE 54276,33:TT=0
1050 POKE 54273,2:POKE 54272,220
1060 FOR DD=1 TO 350:NEXT DD
1070 POKE 54273,3:POKE 54272,103
1080 FOR DD=1 TO 350:NEXT DD
1090 IF TT=4 THEN POKE 54276,0:POKE 54277,0:POKE
    54278,0:TT=0:RETURN

1100 TT=TT+1:GOTO 1050
2000 GOSUB 1030
2010 GOTO 2010
2500 POKE 54296,15:TT=0
2505 POKE 54277,17:POKE 54278,17:POKE 54276,65
2510 POKE 54275,0:POKE 54274,255
2520 POKE 54273,14:POKE 54272,107
2530 FOR DD=1 TO 100:NEXT
2540 POKE 54276,0:POKE 54277,0:POKE 54278,0
2550 FOR DD=1 TO 50:NEXT
2560 IF TT>6 THEN RETURN
2570 TT=TT+1:GOTO 2505
3000 FOR DD=1 TO 2000:NEXT DD:PRINT "□"
3005 POKE 53280,7:POKE 53281,2
3006 PRINT "XXXXXX"
3007 FOR GG=1 TO 40:PRINT "■" CHR$(113);:NEXT GG:
    REM YELLOW
3010 PRINT:PRINT "XX" TAB(12) "YOU HAVE WON!!":
    REM CYAN
3012 PRINT "XX":FOR GG=1 TO 40:PRINT "■" CHR$(113);
    :NEXT GG:REM YELLOW

3020 POKE 54296,15
3030 POKE 54277,34:POKE 54278,130
3040 POKE 54276,17
3050 HF=54273:LF=54272
3060 POKE HF,57:POKE LF,172

```

```

3070 FOR DD=1 TO 125:NEXT DD
3080 POKE HF,64:POKE LF,188
3090 FOR DD=1 TO 125:NEXT DD
3100 POKE HF,51:POKE LF,97
3110 FOR DD=1 TO 125:NEXT DD
3120 POKE HF,57:POKE LF,172
3130 FOR DD=1 TO 500:NEXT DD
3140 POKE 54276,0:POKE 54277,0:POKE 54278,0
3200 GOTO 3200
3500 FOR DD=1 TO 1000:NEXT
3510 PRINT "J":REM CLEAR SCREEN
3520 POKE 53280,0:POKE 53281,2
3530 PRINT "XXXXXXXXXX"
3540 PRINT TAB(8) "##### YOU HAVE BEEN SUNK ###":
      REM YELLOW
3550 GOSUB 1030
3560 GOTO 3560

```

TUNNELS OF THAR

Somewhere in the mysterious Tunnels of Thar lies the long lost Holy Grail. Being a bold Crusader, good and true, you have ventured into these uncharted tunnels, braving the musty blackness with only a medieval Grail Detector to help you. Medieval Grail Detectors weren't very sophisticated so it may take a while to interpret the readout the detector gives you.

At the start of the game the computer will give you a quick look at a plan view of the tunnel system, after this you are on your own. To move through the maze of tunnels, press the first letter of the direction you wish to travel, for instance "N" for north. After each turn the computer will tell you which directions are open to you.

Such as:

NORTH: WALL

SOUTH: OPEN

EAST: OPEN

WEST: WALL

The computer will also give you the readout of your Grail Detector.

If you get lost or confused, pressing "H" for help will reward you with another look at the plan view of the tunnels. Open passages will be shown in yellow, your position is marked with a blue asterisk. Unfortunately, asking for help results in a penalty of 15 moves. Your score at the end of the game depends on how many moves you took to find the Grail.

In this program the routine from 530 sets up the parameters of the program. The lines 550 to 580 decide the position of the Holy Grail. This will change each time the program is RUN.

The DATA statements in lines 750 to 840 provide the locations inside the array for the tunnels. The array is DIMensioned in line 540. Lines 690 to 720 READ the DATA into the array.

Lines 380 to 520 print the plan of the tunnels on the screen. Lines 210 to 330 accept your moves and work out your new position. Lines 50 to 190 provide the feedback printed on the screen after each move and line 190 works out the reading of the Grail Detector. Line 310 checks

to see whether you have found the Grail. If you have, the subroutine from line 1000 congratulates you and tells you how many moves it took you to complete the task.

```

10 REM TUNNELS OF THAR
15 POKE 53281,5:POKE 53280,1
16 PRINT "T":GOSUB 1200:REM CLR SCRN
17 PRINT TAB(12) "TUNNELS OF THAR":REM
    CURSOR DOWN BLACK
20 GOSUB 530
30 GOSUB 370
40 M=M+1
50 PRINT "T"
60 PRINT TAB(5) "MOVE NUMBER:";M:REM CURSOR DOWN
    GREY 2
70 PRINT TAB(5) "NORTH: ";:REM CURSOR DOWN
80 IF A(D+1,E)=S THEN PRINT "OPEN"
90 IF A(D+1,E)=X THEN PRINT "WALL"
100 PRINT TAB(5) "SOUTH: ";:REM CURSOR DOWN
110 IF A(D-1,E)=S THEN PRINT "OPEN"
120 IF A(D-1,E)=X THEN PRINT "WALL"
130 PRINT TAB(5) "EAST: ";:REM CURSOR DOWN
140 IF A(D,E+1)=S THEN PRINT "OPEN"
150 IF A(D,E+1)=X THEN PRINT "WALL"
160 PRINT TAB(5) "WEST: ";:REM CURSOR DOWN
170 IF A(D,E-1)=S THEN PRINT "OPEN"
180 IF A(D,E-1)=X THEN PRINT "WALL"
190 PRINT TAB(5) "GRAIL DETECTOR READS:";100*
    (ABS(Z-D)+ABS(Y-E))+Y-E
210 PRINT TAB(5) "DIRECTION?"
211 PRINT TAB(5) "NORTH SOUTH EAST WEST HELP"
215 GET A$:IF A$="" THEN 215
220 IF A$="N" AND A(D+1,E)=X THEN 215
230 IF A$="S" AND A(D-1,E)=X THEN 215
240 IF A$="E" AND A(D,E+1)=X THEN 215
250 IF A$="W" AND A(D,E-1)=X THEN 215
255 IF A$="H" THEN GOSUB 370
260 IF A$="N" THEN D=D+1
270 IF A$="S" THEN D=D-1
280 IF A$="E" THEN E=E+1
290 IF A$="W" THEN E=E-1

```

```

310 IF Z=D AND Y=E THEN 340
330 GOTO 40
340 GOSUB 1000
350 GOSUB 2000
360 GOTO 360
370 REM HELP
375 POKE 53280,6:POKE 53281,0
380 PRINT "□":PRINT "0":REM CURSOR DOWN
400 PRINT TAB(17) "00NORTH0":PRINT:REM BLUE
    REV ON REV OFF
410 FOR B=15 TO 1 STEP -1:FOR C=1 TO 15
420 PRINT TAB(12);
430 IF A(B,C)=X THEN PRINT "000":REM RED SYMBOL
    FROM COMM KEY AND + KEY
435 IF B=D AND C=E THEN PRINT "0*":GOTO 440:REM
    BLUE
436 IF A(B,C)=S THEN PRINT "0+":REM YELLOW
440 NEXT:PRINT:NEXT
470 PRINT TAB(17) "000SOUTH0":PRINT:REM BLUE
    CURSOR DOWN REV ON REV OFF
480 M=M+15
490 FOR J=1 TO 2000:NEXT
500 PRINT "□":A(D,E)=S:REM CLR SCRN
515 POKE 53280,0:POKE 53281,6
520 RETURN
530 REM INITIALISE
540 DIM A(15,15)
550 B=INT(3*RND(1))+1
560 Z=14:Y=14
570 IF B=2 THEN Y=2
580 IF B=3 THEN Z=2
590 X=1:S=2
610 FOR B=1 TO 15:FOR C=1 TO 15
630 A(B,C)=X:IF RND(1)>.9 THEN A(B,C)=S
640 IF C<20 OR C>14 OR B<20 OR B>14 THEN A(B,C)=X
650 NEXT:NEXT
670 D=2:E=2
690 FOR F=1 TO 68
700 READ B:READ C
710 A(B,C)=S
720 NEXT

```

```

730 M=-15
740 RETURN
750 DATA 2,2,2,3,2,4,2,5,2,6,2,7
760 DATA 3,7,4,7,5,7,5,6,5,5,5,4,5,3,6,3
770 DATA 7,3,7,4,7,5,7,6,7,7,7,8,7,9,9,8
780 DATA 10,8,10,7,10,6,10,5,10,4,8,8
790 DATA 10,3,11,3,12,3,13,3,14,3,14,2,7,10
800 DATA 6,10,5,10,4,10,3,10,2,10,2,11,2,12
810 DATA 2,13,2,14,6,11,6,12,6,13,6,14,7,12
820 DATA 14,12,8,12,8,14,9,12,9,13,9,14,10,12
830 DATA 11,9,11,10,11,11,11,12,12,9,13,9,13,10
840 DATA 13,11,13,12,13,13,13,14,14,14
1000 PRINT "□":REM CLR SCRIN
1010 POKE 53280,4:POKE 53281,11
1020 FOR X=1 TO 4
1030 IF X=1 THEN PRINT "□":REM ORANGE
1040 IF X=2 THEN PRINT "■":REM PURPLE
1050 IF X=3 THEN PRINT "▲":REM CYAN
1060 IF X=4 THEN PRINT "■":REM GREEN
1070 PRINT TAB(12) "X HOLY GRAIL!!!":REM CURSOR DOWN
1080 NEXT
1090 PRINT TAB(8) "X YOU FOUND IT IN";M;"MOVES"
1100 GOSUB 2000
1110 GOTO 1110
1200 POKE 54296,15
1210 POKE 54276,65:POKE 54275,0:POKE 54274,255
1230 POKE 54277,130:POKE 54278,130
1250 POKE 54273,1:POKE 54272,155
1265 FOR X=1 TO 1000:NEXT
1270 POKE 54276,0
1280 POKE 54273,0:POKE 54272,0
1290 RETURN
2000 V=54296:W=54276:A=54277:HF=54273:LF=54272:
    S=54278
2010 POKE V,15
2020 POKE W,65:POKE A,100:POKE 54275,0:POKE
    54274,255
2030 READ AA:IF AA=-1 THEN GOTO 2100
2040 READ BB
2060 POKE HF,AA:POKE LF,BB
2070 POKE S,130
2080 FOR X=1 TO 100:NEXT
2090 GOTO 2020

```

```
2100 POKEHF,34:POKELF,75:FOR X=1TO250:NEXT:POKE W,  
    0:POKE HF,0:POKE LF,0:RETURN  
2110 DATA 51,97,57,172,51,97,38,126,34,75,38,126,  
    43,52,51,97,38,126,34,75,32,94  
2120 DATA 28,214,25,177,28,214,-1,-1
```

CAVERNS OF TERROR

We now come to Caverns of Terror, a game which almost defies description. Most adventure type games are based on logic and are cleverly planned, with many challenges for the player. Difficulties to be overcome, puzzles to solve and usually a final goal to achieve. Caverns of Terror is nothing like that. It is more of an ordeal than an adventure.

Lines 130 to 520 print instructions on the screen, decide whether you are being attacked and if so by what. They keep track of which cavern you are in. They also give you a rather limited choice of defensive weapons. These lines also call up other routines which provide all sorts of odd happenings.

The random number generator in lines 580 to 610 decides whether you have defeated the attacker. Lines 600 to 610 direct the program to the relevant print routine, depending on the result of the generator.

The variable W keeps track of your position in the caverns, G is the amount of gold you are carrying and X records the number of fights you have been in.

The question which now remains unanswered is: "Do you have the nerve, the courage to enter the Caverns of Terror?".

```
10 REM CAVERNS OF TERROR
15 REM "J"=CLEAR SCREEN "M"=DOWN CURSOR
20 PRINT "J":REM CLEAR SCREEN
25 POKE 53280,0:POKE 53281,11
30 PRINT TAB(7) "##### CAVERNS OF TERROR

   ###":REM PURPLE CURSOR DOWN
35 GOSUB 2000
40 GOSUB 1250
50 X=0:G=30:W=1
60 PRINT TAB(1)"M YOU ARE AT THE ENTRANCE TO A
   SYSTEM OF":REM LT RED
```

```

62 PRINT TAB(8)"DARK TUNNELS AND CAVERNS"
65 PRINT TAB(5)"YOU HAVE THIRTY PIECES OF GOLD"
70 PRINT "WHEN YOU REACH THE END OF THE CAVERNS"
72 PRINT "YOU MUST HAVE AT LEAST TWENTY PIECES"
75 PRINT TAB(4)"OF GOLD TO PAY THE GATE KEEPER"
80 PRINT TAB(14) "PRESS RETURN":REM BLACK
90 GET A$
100 IF A$="" THEN GOTO 90
110 GOSUB 910
120 IF W<1 THEN W=INT(8*RND(1))+1
130 PRINT TAB(7)"THIS IS A MAZE OF CAVERNS":
    REM GREEN
150 IF W>=10 THEN GOTO 970
160 PRINT TAB(8)"THIS IS CAVERN NUMBER";W
180 PRINT TAB(6)"CAVERN NUMBER TEN IS THE EXIT"
200 X=X+1
210 PRINT TAB(7)"THIS IS CHALLENGE NUMBER";X
220 IF G<1 THEN G=3
240 PRINT TAB(8)"YOU HAVE";G;"GOLD PIECES"
250 GOSUB 1240
260 GOSUB 910
280 K=INT(4*RND(1))+2
290 PRINT TAB(6)"YOU ARE NOW FACING";K;"
    TUNNELS":REM LT BLUE
300 PRINT TAB(7)"WHICH ONE WILL YOU TAKE";
310 INPUT A
320 GOSUB 910
330 IF RND(1)<.2 THEN GOTO 630
340 IF A<>K THEN 360
350 IF A=K THEN 630
360 K=INT(4*RND(1))+1
370 IF K=1 THEN E$="GHOUL"
380 IF K=2 THEN E$="WEREWOLF"
390 IF K=3 THEN E$="TROLL"
400 IF K=4 THEN E$="VAMPIRE"
410 PRINT TAB(3)"THE TUNNEL IS BLOCKED BY A ";
    E$:REM ORANGE
420 E=INT(4*RND(1))+1
430 IF E=1 THEN F$="POISONED SPEAR"
440 IF E=2 THEN F$="CURSED SWORD"
450 IF E=3 THEN F$="FIREBRAND"
460 IF E=4 THEN F$="CROSSBOW"

```

```

470 PRINT TAB(7)"ARMED WITH A ";F$
500 PRINT TAB(7)"WHICH WEAPON DO YOU CHOOSE:--"
    :REM YELLOW
510 PRINT TAB(7)"A POINTED STICK (1)"
520 PRINT TAB(7)"A SMALL ROCK      (2)"
530 PRINT TAB(7)"BARE HANDS        (3)"
570 INPUT B
580 C=INT(3*RND(1))+1
590 GOSUB 910
600 IF B=C THEN GOSUB 1090
610 IF B<>C THEN GOSUB 1160
620 GOTO 80
630 K=INT(4*RND(1))+1
640 ON K GOSUB 670,720,770,820
650 GOTO 80
670 PRINT TAB(2)"YOU HAVE FALLEN THROUGH A
    TRAPDOOR"
690 W=W-1
700 G=G-INT(2*RND(1))-1
710 RETURN
720 PRINT TAB(5)"A FLASHFLOOD WASHES YOU DOWN"
730 PRINT TAB(13)"A SIDE TUNNEL"
740 W=W+1
750 G=G-INT(2*RND(1))+1
760 RETURN
770 PRINT TAB(10)"YOU HAVE BEEN HELPED"
775 PRINT TAB(12)"BY A PASSING GNOME"
790 G=G+INT(5*RND(1))+1
800 W=W+INT(3*RND(1))+1
810 RETURN
820 PRINT TAB(6)"WHOOPEEE!!! A HOARD OF GOLD"
830 PRINT TAB(4)"YOU MAY TAKE UP TO 5 PIECES ONLY"
840 PRINT TAB(5)"BE CAREFUL, THE MORE YOU TAKE"
850 PRINT TAB(8)"THE MORE IT WILL COST YOU"
860 PRINT TAB(6)"HOW MANY DO YOU WISH TO TAKE";
870 INPUT D
880 IF D>5 THEN GOTO 870
890 G=G+D
900 RETURN
910 PRINT " ":REM CLEAR SCREEN
920 FOR DD=1 TO 250:NEXT
940 RETURN

```

```

960 IF WC>10 THEN RETURN
970 PRINT TAB(5)"YOU ARE STANDING AT THE EXIT"
980 PRINT TAB(7)"DO YOU HAVE ENOUGH GOLD?"
990 PRINT TAB(8)"PRESS RETURN TO FIND OUT"
1000 INPUT C$:PRINT "J"
1010 IF GC<20 THEN PRINT TAB(6)"YOU DO NOT
    HAVE ENOUGH GOLD":REM BLACK
1015 FOR DD=1 TO 1000:NEXT
1016 IF GC<20 THEN PRINT TAB(6)"THE GUARDIAN HAS
    KILLED YOU"
1017 IF GC>=20 THEN GOTO 1040
1020 PRINT TAB(7)"YOU COULD NOT PAY THE TOLL"
1030 GOSUB 2000:END
1040 PRINT TAB(6)"YES, YOU CAN PAY THE TOLL"
    :REM WHITE
1050 PRINT TAB(8)"YOU HAVE WON THE GAME"
1060 PRINT "YOU MAY KEEP THE REMAINING GOLD
    PIECES":REM RIGHT CURSOR
1075 GOSUB 3000
1080 GOTO 1080
1090 PRINT TAB(4)"YOU HAVE FOUGHT YOUR WAY
    FAST"
1095 PRINT TAB(13)"THE ";E$
1100 G=G+INT(3*RND(1))+1
1110 PRINT TAB(8)"AND HAVE";G;"GOLD PIECES"
1120 W=W+INT(3*RND(1))+1
1140 PRINT TAB(7)"YOU ARE APPROACHING CAVE";W
1150 RETURN
1160 PRINT TAB(3)"THE ";E$;" BEAT YOU AND
    LEFT YOU"
1170 G=G-INT(4*RND(1))-1
1180 IF GC<0 THEN G=0
1190 PRINT TAB(8)"WITH";G;"PIECES OF GOLD"
1200 W=INT(8*RND(1))+1
1220 PRINT TAB(5)"AND CHASED YOU INTO CAVERN";W
1230 RETURN
1240 FOR DD=1 TO 3000:NEXT:PRINT "J":RETURN
1250 FOR DD=1 TO 1500:NEXT:PRINT "J":RETURN
2000 POKE 54296,15

```



```
2010 POKE 54277,136:POKE 54278,130
2020 POKE 54276,33
2030 POKE 54273,3:POKE 54272,155
2040 FOR DD=1 TO 500:NEXT
2060 POKE 54273,4:POKE 54272,73
2070 FOR DD=1 TO 750:NEXT
2090 POKE 54273,3:POKE 54272,54
2100 FOR DD=1 TO 1250:NEXT
2110 POKE 54276,0:POKE 54277,0:POKE 54278,0
2120 RETURN
```

REVERSI

This game is based on an old board game very popular in the latter half of the last century.

It is played on an eight by eight checkerboard. The game begins with the centre four locations of the board being occupied with two of your playing pieces (the open circles) and two of the computer's playing pieces (the crosses). Each player takes a turn, placing a new playing piece on the board adjacent to an opponent's piece so that one or more of the opponent's pieces lie between two of your pieces. If this is not done it is considered to be an illegal move and the computer will ignore it. The opponent's pieces trapped between your pieces remain on the board but become your pieces.

The game ends when all the board locations are filled or when neither player can make a legal move. The winner is the player with the most pieces on the board.

At the beginning of the game the computer will ask you whether you wish to have the first move. After you have answered this question, the computer will print the board and playing pieces on the screen. Underneath it will indicate who is having the current turn. When it is your turn to place a piece on the board, enter the location of the square on which you wish to place your circle. This is done by using the square's number down the side of the board followed by the number across the board. Enter this number as a two digit number such as 45.

The computer will keep track of the score and automatically replace captured pieces with the opponent's pieces. You will find that the computer plays this game slowly but it will play very well. In fact you may find it difficult to win a game against the amazing Commodore 64.

```
10 REM REVERSI
15 PRINT "I":GOSUB 6000:REM CLEAR SCREEN
20 PRINT "I":REM CLEAR SCREEN
30 X=ASC("X"):O=ASC("O"):REM LETTER O NOT ZERO
40 DIM A(10,10)
50 FOR B=1 TO 10:FOR C=1 TO 10
```

```

60 IF B<>1 AND C<>1 AND B<>10 AND C<>10 THEN A(B,C)
    =ASC(",")
70 NEXT C:NEXT B
80 A(5,5)=X:A(6,6)=X
90 A(6,5)=0:A(5,6)=0
100 P=0
120 PRINT TAB(9) "WHO DO YOU WANT TO GO FIRST":
    REM YELLOW
130 PRINT TAB(12) "X(YES-1   NO-2)";
140 INPUT W
150 PRINT "I":REM CLEAR SCREEN
155 POKE 53280,0:POKE 53281,11
160 GOSUB 3000
170 IF W=1 THEN 2000
1000 PRINT TAB(16) "MY MOVE":REM REV ON CYAN
1010 S=0
1020 T=X
1030 H=0
1040 FOR A=2 TO 9
1050 FOR B=2 TO 9
1060 IF A(R,B)<>46 THEN 1320
1070 Q=0
1080 FOR C=-1 TO 1
1090 FOR D=-1 TO 1
1100 K=0
1110 F=A
1120 G=B
1130 IF A(F+C,G+D)<>S THEN 1180
1140 K=K+1
1150 F=F+C
1160 G=G+D
1170 GOTO 1130
1180 IF A(F+C,G+D)<>T THEN 1200
1190 Q=Q+K
1200 NEXT D
1210 NEXT C
1220 IF A=2 OR A=9 THEN Q=Q*2
1230 IF B=2 OR B=9 THEN Q=Q*2
1240 IF A=3 OR A=8 THEN Q=Q/2
1250 IF B=3 OR B=8 THEN Q=Q/2
1260 IF (A=2 OR A=9) AND (B=3 OR B=8) THEN Q=Q/2
1270 IF (A=3 OR A=8) AND (B=2 OR B=9) THEN Q=Q/2

```

```

1280 IF Q<H OR Q=0 OR (RND(1)>.3 AND Q=H)
    THEN 1320
1290 H=Q
1300 M=A
1310 N=B
1320 NEXT B
1330 NEXT A
1340 IF H=0 AND R=0 THEN 5000
1350 IF H=0 THEN 1370
1360 GOSUB 4000
1370 GOSUB 3000
2000 PRINT TAB(14) "■YOUR TURN":REM BLACK
2005 INPUT R
2007 FOR XX=1784 TO 1863:POKE XX,32:NEXT
2010 S=X
2020 T=0:REM LETTER 0
2035 REM ZERO TO PASS
2040 IF R=0 THEN 2090
2050 IF R<11 OR R>88 THEN 2000
2060 M=INT(R/10)+1
2070 N=R-10*INT(R/10)+1
2080 GOSUB 4000
2090 GOSUB 3000
2100 GOTO 1000
3000 PRINT "■":REM HOME
3010 C=0
3020 H=0
3030 PRINT:PRINT TAB(16) "■REVERSI":PRINT:REM
    REV ON GREEN
3035 PRINT TAB(15) "■12345678":REM LT RED
3040 FOR F=2 TO 9
3050 PRINT "■"TAB(12) F-1:REM LT RED
3060 FOR G=2 TO 9
3070 PRINT "■" CHR$(A(F,G)):REM LT BLUE REV ON
3075 PRINT "■";
3080 IF A(F,G)=X THEN C=C+1
3090 IF A(F,G)=O THEN H=H+1
3100 NEXT G
3110 PRINT "■" F-1:REM LT RED
3120 NEXT F
3130 PRINT:PRINT TAB(15) "■12345678":REMLT RED
3140 PRINT
3145 POKE 1693,32

```

```

3150 PRINT TAB(10) "THE COMPUTER HAS:";C:REM
      YELLOW
3155 PRINT
3157 POKE 1772,32
3160 PRINT TAB(12) "THE HUMAN HAS:";H:REM ORANGE
3165 PRINT
3170 RETURN
4000 FOR C=-1 TO 1
4010 FOR D=-1 TO 1
4020 F=M
4030 G=N
4040 IF A(F+C,G+D)<>S THEN 4080
4050 F=F+C
4060 G=G+D
4070 GOTO 4040
4080 IF A(F+C,G+D)<T THEN 4140
4090 A(F,G)=T
4100 IF M=F AND N=G THEN 4140
4110 F=F-C
4120 G=G-D
4130 GOTO 4080
4140 NEXT D
4150 NEXT C
4160 RETURN
5000 FOR B=2 TO 9
5010 FOR I=2 TO 9
5020 IF A(B,I)=X THEN C=C+1
5030 IF A(B,I)=0 THEN H=H+1
5040 NEXT I:NEXT
5045 FOR ID=1 TO 2000:NEXT:PRINT " ":REM CLEAR
      SCREEN
5047 POKE 53280,11:POKE 53281,0
5049 PRINT "XXXXXXXXXXXX":REM DOWN CURSOR
5050 IF C>H THEN PRINT TAB(4) "THE COMPUTER HAS
      WON!!!";C;"TO";H:REM YELLOW
5060 IF H>C THEN PRINT TAB(6) "THE HUMAN HAS WON";
      H;"TO";C
5070 IF H=C THEN PRINT TAB(14) "IT IS A DRAW"
5080 END
6000 POKE 53280,10:POKE 53281,2

```

```
6010 PRINT "|||||":REM CURSOR DOWN
6020 PRINT TAB(16) "REVERSI":REM REV ON BLACK
    REV OFF
6030 FOR DD=1 TO 1500:NEXT DD
6040 RETURN
```

COMMODORE CHECKERS

Here is a simple Checkers program which you will find is a lot of fun to play. The game follows traditional rules except that there is no penalty for refusing to capture an opponent's piece. However the computer will jump one of your playing pieces whenever it can.

The computer's pieces are the solid circles playing down from the top of the board and your pieces are the CLUB symbols playing up from the bottom. Kings are made automatically. The computer's kings are shown by a diamond shape and yours by the letter K.

To move a piece, enter the location of the piece you wish to move when the computer asks "MOVE FROM". This is entered as a letter followed by a number, such as G6. Likewise enter the location of the square you wish to move to when the computer asks "MOVE TO". You will see that the board is printed on the screen with the letters ABCDEFGH across the top and bottom and numbers down the side to assist you.

After you have jumped one of the computer's pieces, you will be asked "JUMP AGAIN (Y/N)". Enter Y if you can jump again and N if you can't.

There is no mechanism to prevent you from cheating but there is no point in doing so. A piece which is moved illegally will confuse the program and such a piece may tend to disappear from the board entirely.

The computer always has the first move. The number of pieces captured is shown above the board after each turn. The game ends when either one of you has lost all his pieces or if the computer decides it is in a hopeless position and concedes the game.

Strategy is not the computer's greatest virtue, but it will put up a spirited defence when under attack and it will be very reluctant to move into danger.

The routine from line 950 to 1030, called up by line 50 at the beginning of the game, sets up the starting conditions. It sets up an array which will hold the information required for the playing board. Line 960 assigns the value of 9 to all the elements of the array. This value represents the squares which aren't used during play, that is the black squares. The next line, 970, reads the contents of the DATA statements

into 32 elements of the array. These numbers represent the 32 white squares used for playing the game. This Checkers program uses a system of numbered squares based on a system devised by A. L. Samuels. The white squares are numbered from the top left hand corner down to the bottom left hand corner. The top row is numbered 72, 71, 70, 69 and the next row is numbered 66, 65, 64, 63. The next row is 59, 58, 57, 56. The next 53, 52, 51, 50 and so on. This numbering system means that the computer can keep track of the moves by the numerical difference between the starting and finishing square. Moving up and to the right always leaves a difference of plus six, moving down and to the right minus six, moving up and to the left gives a difference of plus seven and down and to the left, a difference of minus seven. In this way any move on the board will result in a predictable number which makes the whole system very easy for a computer to handle.

The DATA statements consist of pairs of numbers. The first two-digit number, e.g. 72, represents the number of a square, the second number the value of the square. The value of minus one means that the square is occupied by one of the computer's playing pieces, a zero means it is empty and the value of plus one indicates the human player's pieces. The computer's kings are minus 2 and the human player's kings plus 2.

Subroutine 740 to 920 uses this information to print the board on the screen.

The routine called up by line 740 converts the values of the squares into CHR\$ codes to print up the correct piece on the board. For instance CHR\$(32) is a blank space and CHR\$(113) is a solid circle used for the computer's playing pieces. By changing these values you can alter the shapes of the playing pieces.

Subroutine 2000 converts the CHR\$ codes back into the original values so that they can be used in the calculations for the computer's move.

The computer uses lines 390 to 405 to check the safety of its move. If after 1000 checks (line 430) it can't find a safe move it will then concede the game.

Lines 130 to 180 check to see whether the computer can jump one of your pieces. Lines 480 to 490 accept your move. Line 490 clears the information from the screen ready for the next turn.

Lines 540 to 576 check to see which square you have moved from and the new square you have moved to. They convert the computer's numbering system to the one used to display the square numbers on the screen.

```

10 REM CHECKERS
15 REM "X" IS DOWN CURSOR
20 POKE 53280,0:POKE 53281,7
30 PRINT "O":REM CLEAR SCREEN
40 PRINT "XXXXXXXX" TAB(12) "PLEASE STAND BY":
    REM RED
50 GOSUB 950
60 Z=24
70 Q=0
90 FOR G=69 TO 72:IF A(G)=-1 THEN A(G)=-2
100 NEXT
105 POKE 53280,2:POKE 53281,6
110 GOSUB 740
130 IF A(Z)=9 OR A(Z)<1 THEN GOTO 220
140 IF Z<28 AND A(Z)=1 THEN A(Z)=2
150 Y=1
160 IF A(Z+X(Y))<0 AND A(Z+2*(X(Y)))=0 THEN Q=X(Y)
170 IF A(Z)=2 AND A(Z-X(Y))<0 AND A(Z-2*(X(Y)))=0
    THEN Q=-X(Y)
180 IF Q>0 AND Z+2*Q>23 THEN GOTO 250
190 Q=0
200 IF Y=2 THEN GOTO 220
210 Y=2:GOTO 160
220 Z=Z+1
230 IF Z<73 THEN 130
240 IF Q=0 THEN 340
250 A(Z+Q)=0:A(Z+2*Q)=A(Z):A(Z)=0
260 Z=Z+2*Q:CO=CO+1:GOSUB 740
270 Q=0:Y=1
280 IF A(Z+X(Y))<0 AND A(Z+2*(X(Y)))=0 THEN Q=X(Y)
290 IF Q>0 AND Z+2*Q>23 THEN 250
310 IF Y=1 THEN Y=2:GOTO 280
320 GOTO 480
340 U=0:Q=0
350 Z=24+INT(RND(1)*49):U=U+1
360 IF(A(Z)=9 OR A(Z)=-1 OR A(Z)=-2 OR A(Z)=0) AND
    U<1000 THEN GOTO 350

```

```

380 Y=1
385 IF A(Z+X(Y))<>0 THEN 420
390 IF A(Z+X(Y))=0 AND A(Z+2*X(Y))>-1 AND A(Z+2*X
(Y)+1)>-1 THEN Q=X(Y)
395 IF A(Z+X(Y))=0 AND U>150 THEN Q=X(Y)
400 IF A(Z)=2 AND A(Z-X(Y))=0 AND A(Z-2*X(Y)+1)>-1
THEN Q=-X(Y)
405 IF U>600 AND A(Z)=2 AND A(Z-X(Y))=0 THEN
Q=-X(Y)
410 IF Q<>0 THEN GOTO 450
420 IF Y=1 THEN Y=2:GOTO 390
430 IF U<1000 THEN GOTO 350
440 PRINT "M" TAB(10) "I CONCEDE THE GAME":END
450 A(Z+Q)=A(Z):A(Z)=0
470 GOSUB 740
480 PRINT:PRINT TAB(8) "MOVE FROM":REM PURPLE
482 INPUT A$
485 PRINT "M" TAB(8) "MOVE TO":
487 INPUT B$
490 FOR GG=1743 TO 1863:POKE GG,32:NEXT GG
510 FOR W=1 TO 2:Z=0
520 IF W=1 THEN C$=A$
530 IF W=2 THEN C$=B$
540 Z=-24*(C$="G9")-25*(C$="E9")-26*(C$="C9")
542 IF Z<>0 THEN GOTO 580
544 Z=-27*(C$="A9")-30*(C$="H8")-31*(C$="F8")
546 IF Z<>0 THEN GOTO 580
548 Z=-32*(C$="D8")-33*(C$="B8")-37*(C$="G7")
550 IF Z<>0 THEN GOTO 580
552 Z=-38*(C$="E7")-39*(C$="C7")-40*(C$="A7")
554 IF Z<>0 THEN GOTO 580
556 Z=-43*(C$="H6")-44*(C$="F6")-45*(C$="D6")
558 IF Z<>0 THEN GOTO 580
560 Z=-46*(C$="B6")-50*(C$="G4")-51*(C$="E4")
562 IF Z<>0 THEN GOTO 580
564 Z=-52*(C$="C4")-53*(C$="A4")-56*(C$="H3")-57*
(C$="F3")
566 IF Z<>0 THEN GOTO 580
568 Z=-58*(C$="D3")-59*(C$="B3")-63*(C$="G2")
570 IF Z<>0 THEN GOTO 580

```

```

572 Z=-64*(C$="E2")-65*(C$="C2")-66*(C$="A2")-69*
    (C$="H1")-70*(C$="F1")
574 IF Z<>0 THEN GOTO 580
576 Z=-71*(C$="D1")-72*(C$="B1")
580 IF W=1 THEN D=Z
590 IF W=2 THEN E=Z
600 NEXT A$:B$="":C$=""
620 A(E)=A(D):A(D)=0
650 IF ABS(D-E)>7 THEN A((D+E)/2)=0:HU=HU+1
670 GOSUB 740
680 IF ABS(D-E)>7 THEN PRINT "X" TAB(4) "JUMP
    AGAIN (Y/N)";
685 IF ABS(D-E)>7 THEN INPUT U$:IF U$<>"N"
    THEN GOTO 480
687 FOR GG= 1744 TO 1783:POKE GG,32:NEXT GG
690 IF HU<12 AND CO<12 THEN GOTO 60
700 IF HU=12 THEN PRINT "X" TAB(4) "THE HUMAN
    WINS":END
710 IF CO=12 THEN PRINT "X" TAB(6) "I WIN I WIN!!!"
740 GOSUB 1500
770 PRINT "X":PRINT "X" TAB(8) "COMPUTER: ";CO;
    "HUMAN: ";HU:REM HOME WHITE
780 PRINT:PRINT:PRINT
785 PRINT TAB(14) "ABCDEFGH":REM REV ON GREEN
790 T=-2:FOR K=0 TO 3:PRINT TAB(14):FOR J=0 TO 3
800 PRINT "███"; CHR$(A(72-J-13*K)):NEXT:T=T+1:
    REM BLACK (COMMKEY +) CYAN
820 PRINT "█" INT((J+K)/2)+T:REM YELLOW
830 FOR J=0 TO 3:PRINT TAB(14) "███" CHR$(A(66-J-13
    *K)):PRINT:REM CYAN BLACK
850 NEXT:T=T+1
860 PRINT "█" INT((J+K)/2)+T:REM YELLOW
870 NEXT
880 PRINT TAB(14)"ABCDEFGH":PRINT:REM REV ON
    GREEN
890 GOSUB 2000
920 RETURN
950 DIM A(99):X(1)=-6:X(2)=-7

```

```

960 FOR Z=1 TO 99:A(Z)=9:NEXT
970 FOR Z=1 TO 32:READ B:READ C:A(B)=C:NEXT
980 DATA 72,1,71,1,70,1,69,1,66,1,65,1,64,1,63,1,
      59,1,58,1,57,1,56,1
990 DATA 53,0,52,0,51,0,50,0,46,0,45,0,44,0,43,0
1000 DATA 40,-1,39,-1,38,-1,37,-1,33,-1,32,-1,31,
      -1,30,-1,27,-1,26,-1,25,-1
1005 DATA 24,-1
1010 CO=0:HU=0
1020 PRINT "□":REM CLEAR SCREEN
1030 RETURN
1500 FOR M=24 TO 72
1510 IF A(M)=1 THEN A(M)=209
1520 IF A(M)=2 THEN A(M)=113
1530 IF A(M)=0 THEN A(M)=32
1540 IF A(M)=-1 THEN A(M)=120
1550 IF A(M)=-2 THEN A(M)=75
1560 NEXT
1570 RETURN
2000 FOR M= 24 TO 72
2010 IF A(M)=209 THEN A(M)=1
2020 IF A(M)=113 THEN A(M)=2
2030 IF A(M)=32 THEN A(M)=0
2040 IF A(M)=120 THEN A(M)=-1
2050 IF A(M)=75 THEN A(M)=-2
2060 NEXT
2070 RETURN

```

BLOCKOUT

This is a very impressive, fast moving game which will test your reactions and skill. The game of Blockout can be played by one or two people. The single player game can be played from the keyboard, but in the two player game one player will need a joystick which should be plugged into port number two.

In the one player game you begin near the left hand side of the screen and rush off towards the opposite side of the playing area, leaving a trail behind you. The object of the game is to keep moving for as long as possible without hitting a wall or your own trail. In the two player game you must avoid your opponent's trail as well as your own while trying to trap him into making a mistake. Player one leaves a red trail and player two, starting near the right of the screen, leaves a blue trail.

In the two player game, player one uses the keyboard for controlling his man. The "A" key moves him up the screen and the "X" key moves him down the screen. The "," key and the "." move him left and right respectively. Player two uses the joystick plugged into port number two. In the one player game either the keyboard or a joystick can be used.

At the beginning of the game the function keys are used to select the game options. Function key number one is used to clear out the previous high score if you wish to do this. Key number three chooses the type of game you wish to play. The options are two players, one player with joystick or one player with keyboard. Key number five is used to select how many game turns you wish to play. When you have the options you want displayed on the screen, press function key seven to start the game. If you wish to finish playing, pressing function key seven will return you to the title screen.

Line 15 prints up the playfield. Lines 20 to 27 read the keyboard and print the trail for player one. X1%\$ is the variable which controls the screen position of player one. Lines 30 to 37 read the input from the joystick and print the trail for player two using variable X2%\$.

The function keys are read by lines 40 to 60. These keys are used to select the game options and start and finish the game. Lines 65 and 67 use P\$ to place a string of characters onto the screen. T% keeps track of the location of P\$.

The routine in lines 70 to 80 is used to provide a precise time delay using TI\$. This routine also checks to see whether function key seven has been pressed. If so, the game is terminated and the title screen is displayed.

The collisions between players or a player and the wall are detected by lines 85 to 100. If there are two players, this routine allows any unaffected player to continue on until he has a collision.

The scores are printed up by lines 105 to 115 and lines 120 to 135 tell you which player has won the game and also display the bonus points.

Lines 140 to 150 check to see if the game is in one or two player mode and return the program to the appropriate line to start the loop again.

Line 155 sets up the variables for the start of the game and line 160 calls up the subroutine which prints the scores. The title is printed by the PRINT lines in 161 to 185. The arrays and variables are initialised in lines 205 to 227.

This very clever program was written by Gary Ryan of Canberra.

```

5 REM BLOCKOUT, BY G. RYAN
10 GOTO 205
15 PRINT BR$:FOR Z=1.T0.21:PRINT "  ";TAB(38);"  "
   :NEXT:PRINT BR$;"  ":RETURN
16 REM "  " IS ON + KEY "  " IS HOME CURSOR
20 GET K$:DX=ASC(K$+Z$):PX=1:BZ=BZ+1:PRINT
   "XXXXXXXXXXXXXXXX":BZ
21 REM HOME LT RED DOWN CURSOR RIGHT CURSOR
22 IF KX(DX) THEN LKX=KX(DX)
25 X1X=X1X+LKX:TAX=SMX+X1X:LCX=PRX-(PEEK(TAX)=32)
   :POKE TAX,C1X
27 POKE CM+X1X,L1X:ON LCX GOTO90,20,85,30
30 DX=PEEK(J) AND15:PX=2:BZ=BZ+1:PRINT
   "XXXXXXXXXXXXXXXX":BZ:REM SAME AS LINE 20
32 IF JX(DX) THEN LJX=JX(DX)
35 X2X=X2X+LJX:TAX=SMX+X2X:LCX=PRX-(PEEK(TAX)=32)
   :POKE TAX,C2X

```

```

37 POKE CM+X2%,L2%:ON LC% GOTO90,30,85,20
40 GET K$:YZ=ASC(K#+CHR$(0)):GZ=1-((XZ=134)+2*
    (XZ=135)+3*(XZ=133)+4*(XZ=136))
43 ON G% GOTO 40,45,50,55,60
45 PLZ=PLZ+1:PLZ=-(PLZ<3)*PLZ:TZ=1809:P#=PL$(PLZ)
    :CZ=13:GOSUB 65:GOTO 40
50 RZ=RZ+1:RZ=-(RZ<5)*RZ:TZ=1894:RMZ=RNZ(RZ):
    P#=STR$(RMZ)+" "
52 CZ=13:GOSUB 65:GOTO 40
55 HSZ=0:TZ=1081:P#="" 00 " :CZ=10:GOSUB65:GOTO 40
60 RETURN
65 FOR Q=1 TO LEN(P#):AZ=ASC(MID$(P#,Q,1)):POKE
    TZ+Q-1,AZ+64*(AZ>63)
67 POKE CM-1024+TZ+Q-1,CZ:NEXT :RETURN
70 GET K$:IF ASC(K#+CHR$(0))=136 THEN 160
75 IF VAL(TI#)<>T THEN 70
80 RETURN
85 PRZ=1:ON P% GOTO 30,20
90 SZ(PZ)=SZ(PZ)+BZ:IF SZ(1)>HSZ THEN HSZ=SZ(1)
95 IF SZ(2)>HSZ THEN HSZ=SZ(2)
100 TI#="000000":T=3:GOSUB 70:GOTO 120
105 PRINT"┌─┐ PLR <1>┐ BONUS┐ HI SCR┐ ROUND┐
    PLR <2>"
106 REM CLEAR SCREEN LT GREEN YELLOW ORANGE BLUE
    LT GREEN
107 LZ=4:DZ=SZ(1):GOSUB 115
110 LZ=20:DZ=HSZ:GOSUB 115:LZ=27:DZ=RNZ:GOSUB 115
112 LZ=35:DZ=SZ(2):GOSUB 115:RETURN
115 PRINT "┌─┐"TAB(LZ-INT(LEN(STR$(DZ))/2))DZ:
    RETURN
120 GOSUB 105:PRINT "┌─┐"TAB(11);"┌─┐BONUS FOR
    ROUND";RNZ;"┌─┐"
121 REM HOME┐ LT RED DOWN CURSOR LT BLUE
122 PRINT TAB(12);"GOES TO PLAYER";P%
125 PRINT "┌─┐"PRINT TAB(15)BZ" POINTS":TI#="
    "000000":T=4:GOSUB 70
126 REM LT GREEN DOWN CURSOR

```

```

130 BZ=100:RNZ=RNZ+1:LKZ=1:LJZ=-1:IF RNZ> RMZ
    THEN 200
135 X1Z=528:X2Z=551:GOSUB 105:GOSUB 15:ON PLZ
    GOTO 140,145,150
140 PRZ=3:GOTO 20
145 PRZ=1:GOTO 30
150 PRZ=1:GOTO 20
155 SZ(1)=0:SZ(2)=0:RNZ=0:PLZ=PLZ+1:GOTO 130
160 PLZ=0:RX=0:RNZ=0:GOSUB 105
161 PRINT"*****"

```

*****:REM HOME DOWN PURPLE

```

162 PRINT" ";TAB(38);""
165 PRINT"  "
167 PRINT"  "
170 PRINT"  "
172 PRINT"  "
175 PRINT"  "
180 PRINT"  "
182 PRINT"  "TAB(38);""
185 PRINT" *****BY G.RYAN, 1983*****"
187 PRINT"X00  F1  CLEAR HIGH SCORE"
188 REM DOWN CURSOR ORANGE REV ON LT BLUE REV OFF
190 PRINT"X  F3  PLAYER OPTIONS: 2 PLAYERS"
192 PRINT"X  F5  SELECT NO. OF ROUNDS: 5"
195 PRINT"X  F7  START GAME/ABORT GAME":
    GOSUB 40:GOTO 240
200 PRINT"*****"TAB(26)"GAME OVER":
    TI$="000000":T$="09":GOSUB 70
201 REM HOME CURSOR DOWN YELLOW
202 PLZ=0:RX=0:RNZ=0:GOTO 160
205 DIM KZ(255),JZ(15),RNZ(4),SZ(2),PL$(2):
    JZ(14)=-40:JZ(13)=40:JZ(5)=1
207 JZ(6)=1:JZ(7)=1:JZ(9)=-1:JZ(10)=-1:JZ(11)=-1
210 KZ(65)=-40:KZ(97)=-40:KZ(83)=-40:KZ(115)=-40:
    KZ(62)=1:KZ(46)=1
212 KZ(89)=40:KZ(120)=40:KZ(67)=40:KZ(99)=40:
    KZ(44)=-1:KZ(60)=-1
215 RNZ(0)=5:RNZ(1)=10:RNZ(2)=20:RNZ(3)=50:
    RNZ(4)=100

```


STOCK EXCHANGE

Your Commodore 64 will now give you a chance to make a fortune on the stock market. Unfortunately if you manage to make a million dollars you will find it very difficult to spend because your computer only plays with computer dollars which can't be spent outside the program. The good news is, of course, that if you go bankrupt playing this game the computer can't make you pay off your debts.

You begin the game with a limited amount of money and you have to manage this money carefully, buying and selling shares, to set yourself up as a successful investor.

The computer will display a balance sheet on the screen. This will show you the status of four companies, how many shares you hold in each company, the price per share, the price you paid for the shares if you hold any and the net profit or loss on your investment. The headings for these columns are provided by lines 160 and 170. Type these lines in exactly as they are shown in the listing, including the spaces.

If you attempt to sell shares you do not hold the computer will charge you with FRAUD. If you try to spend more money than you have you will be declared bankrupt.

As in real life, your whole share portfolio can be rendered worthless by a sudden collapse in the share market.

```
10 REM STOCK EXCHANGE
15 GOSUB 600
20 POKE 53280,6:POKE 53281,2
30 PRINT "I":REM CLEAR SCREEN
50 X=4:I=10:B=1000
100 DIM A(3,5)
110 FOR A=1 TO X
115 A(1,A)=INT(200*RND(1))+A
120 A(3,A)=0
130 NEXT A
140 PRINT "I":REM CLEAR SCREEN
160 PRINT "XXXXCO.    YOU    PRESENT    ORIGINAL
      PROFIT":REM BLACK
```

```

170 PRINT "NO.    HOLD    PRICE    PRICE    /LOSS"
180 FOR A=1 TO X
190 Z=A(1,A)-A(3,A)
200 PRINT "■" A;TAB(X+2);A(2,A);TAB(13) A(1,A);
      TAB(24) A(3,A);TAB(33) Z
202 REM YELLOW
210 NEXT A
230 PRINT "■BANK";B;"INTEREST RATE";I
231 REM LT BLUE
240 PRINT "■WHICH COMPANY (1-4)";
245 INPUT A
250 PRINT:PRINT "■HOW MANY SHARES - A NEGATIVE
255 INPUT C
270 IF C>0 THEN A(3,A)=A(1,A)
280 B=B-C*A(1,A)
290 IF B<0 THEN GOTO 410
300 A(2,A)=A(2,A)+C
310 IF A(2,A)<0 THEN GOTO 500
320 FOR A=1 TO X
330 A(1,A)=INT(ABS(A(1,A)+A*50*(RND(1)-RND(1))))
340 NEXT A
350 I=INT(ABS(10+RND(1)*8-X)+1)
360 B=INT(B+B*I/100)
365 IF B>999999 THEN GOTO 700
370 IF RND(1)>.05 THEN GOTO 140
375 FOR DD=1 TO 1000:NEXT DD
380 PRINT "■":POKE 53280,0:POKE 53281,6:REM
      CLEAR SCREEN.
385 PRINT "■"TAB(5) "■THE STOCK MARKET HAS
      COLLAPSED":REM YELLOW
390 GOTO 390
410 FOR DD=1 TO 1000:NEXT DD
420 POKE 53280,2:POKE 53281,10
430 PRINT "■"TAB(5):REM CLEAR SCREEN CURSOR DOWN
440 PRINT TAB(6) "■#####  BANKRUPT  #####":
      REM BLUE
450 GOTO 450
500 FOR DD=1 TO 1000:NEXT
505 PRINT "■":REM CLEAR SCREEN
510 POKE 53280,11:POKE 53281,0
520 PRINT "■"TAB(5):REM CURSOR DOWN

```

```

530 PRINT TAB(10) "XXXXX FRAUD XXXXX":REM PURPLE
540 GOTO 540
600 PRINT "J":REM CLEAR SCREEN
610 POKE 53280,4:POKE 53281,5
620 PRINT "XXXXXXXXXX":REM CURSOR DOWN
630 PRINT TAB(6) "XXXXXX STOCK EXCHANGE XXXXX"
640 FOR TT=0 TO 15
650 POKE 53280,TT
660 FOR DD=1 TO 100:NEXT DD
670 NEXT TT
680 RETURN
700 FOR DD=1 TO 1000:NEXT DD
710 PRINT "J":REM CLEAR SCREEN
720 POKE 53280,5:POKE 53281,4
730 PRINT "XXXXXXXXXX":REM CURSOR DOWN
740 PRINT TAB(6) "XXX YOU ARE A MILLIONAIRE XXX":
    REM BLUE
750 GOTO 750

```

EARTH BASE ONE

Your duty is to defend earth from the alien invaders riding gravity beams down to the surface of our planet. Your weapon is a laser cannon and you must hit each alien before it reaches the ground.

To move your laser cannon left and right use the “L” key and the “;” key respectively. The “A” key fires the cannon.

When you hit the first alien, two more attack. Remove these two and three begin to descend. This continues until all of your 5 lives are used up. You lose a life each time an alien reaches the ground.

Line 50 DIMs the arrays used in the program. Line 63 and line 65 set the starting positions for the aliens. Line 70 is the starting co-ordinates for your laser cannon. Line 80 starts the alien off on its journey to earth. After you hit an alien, line 90 checks to see whether there are any other aliens still on the screen. If there are, the loop continues dropping the remaining aliens down the screen. If the array is empty, the program goes to line 250 to start off the next wave.

Lines 100, 105, 120 use the statement PEEK (197) to read the input from the keyboard. Line 105 checks to see whether you are moving your laser cannon left (K=42) or right (K=50). Line 120 checks to see whether you are firing your cannon (PEEK(197)=10).

Variable Y is the location of your missile. Line 130 checks the positions of the aliens to see whether they have collided with a missile. Line 140 alters the position of the missile and checks to see whether it has hit an alien.

The routine beginning at line 500 and running to line 580 controls the explosion when an alien is hit. Line 500 removes the alien from the array. Lines 510 to 520 POKE the explosion onto the screen and line 540 increments your score. Line 550 turns the explosion back, which creates the very effective white flash followed by the black aftermath of the explosion.

Earth Base One is based on a program by Matthew Boek.

```

5 REM EARTH BASE ONE
7 PRINT "J":REM CLEAR SCREEN
9 POKE 53280,0:POKE 53281,11
10 PRINT "XXXXXXXXXX" TAB(8) "J### EARTH BASE ONE
    ###":REM ORANGE
20 PRINT "XXXXXX" TAB(9) "PRESS ANY KEY TO BEGIN"
30 GET A$:IF A$="" THEN 30
40 PRINT "J":REM CLEAR SCREEN
50 DIM A(20),AZ(20):N=1:LI=5
60 PRINT "JH";TAB(5) "SCORE:":S;TAB(25) "LIVES:":
    LI:IF N>20 THEN N=20
61 REM CLEAR SCREEN HOME REV ON PURPLE
63 FOR I=1 TO 10:A(I)=1070+(I*2):AZ(I)=1:NEXT:H=0:
    NL=N:B=0
65 FOR I=11 TO 20:A(I)=1186+(I*2):AZ(I)=1:NEXT
70 X=1954:CO=54272
80 FOR I=1 TO N:POKE A(I),88:POKE A(I)+CO,1:NEXT
90 FOR SH=1 TO N:IF AZ(SH)=0 THEN 250
95 FOR U=1 TO 100/NL:NEXT
100 K=PEEK(197)
105 IF K<>64 THEN X1=X:X=X+(K=42ANDX>1944)-(K=50
    ANDX<1983):POKE X1,32
110 POKE X,65:POKE X+CO,0
120 IF PEEK(197)=10 AND B=0 THEN Y=X-40:POKE Y,93:
    POKE Y+CO,0:B=1
130 POKE A(SH),32:A(SH)=A(SH)+40:IF PEEK(A(SH))=93
    THEN 500
132 IF A(SH)>1944 THEN 1000
135 POKE A(SH),88:POKE A(SH)+CO,1
140 IF B=1 THEN Y1=Y:Y=Y-40:IF PEEK(Y)=88 THEN 500
150 IF B=1 THEN POKE Y,93:POKE Y+CO,1:POKE Y1,32
160 IF B=1 THEN IF Y<1103 THEN B=0:POKE Y,32
250 NEXT:FOR DD=1 TO 15:NEXT:GOTO 90
500 FOR G=1 TO 20:IF A(G)=Y THEN AZ(G)=0
505 NEXT
510 C=Y+CO:POKE Y,160:POKE C,1:POKE Y-1,160:
    POKE C-1,1:POKE Y+1,160:POKE C+1,1
520 POKE Y+40,160:POKE C+40,1:POKE Y-40,160:
    POKE C-40,1
540 S=S+50

```

```

550 POKE C,0:POKE C-1,0:POKE C+1,0:POKE C+40,0:
    POKE C-40,0
570 NL=NL-1:B=0:H=H+1:IF H=N THEN N=N+1:GOTO 60
580 GOTO 250
1000 PRINT "XXXXXXXXXX":REM HOME DOWN CURSOR
1010 PRINT TAB(10) "GREEN ALIEN HAS LANDED":REM
    REV ON YELLOW REV OFF
1020 LI=LI-1:IF LI=0 THEN 1100
1030 FOR DD=1 TO 2000:NEXT DD:GOTO 60
1100 FOR DD=1 TO 1500:NEXT DD
1110 PRINT "XXXXXXXXXX"
1120 PRINT TAB(10) "THE GAME IS OVER":REM
    REV ON GATNEEN REV OFF
1130 FOR DD=1 TO 1500:NEXT DD
1140 PRINT TAB(11) "WANT ANOTHER GAME (Y/N)":REM
    ORANGE
1150 POKE 198,0
1160 GET A$:IF A$="" THEN GOTO 1160
1170 IF A$="Y" THEN RUN
1180 PRINT " ":END:REM CLEAR SCREEN

```

ASTEROID

You are the administrator of a mining colony which has been established on a desolate piece of space rock deep inside the asteroid belt. Your isolated community exists by mining the rare ore at the core of the asteroid and selling it to earth in order to buy the food and oxygen supplies your people need.

You must manage the affairs of the colony so that it produces enough ore to survive, but you must not waste your scarce resources doing so.

You will be given a computer print out telling you all the information you need, including the level of your food and oxygen supplies and the cost of maintaining your colony per year. By mining the optimum amount of ore you should be able to survive on the food and oxygen the income will allow you to buy.

Apart from rare attacks from claim jumpers and other assorted bad guys, this game isn't based on random factors or chance. It will involve a great deal of management skill to survive for any length of time. Each time the game is run the price of the ore, oxygen and food units will be different. This prevents the game from becoming predictable.

A look at the program listing will show that the program is quite straight forward. The yearly printout begins at line 120. Lines 80 to 110 provide warnings of potential problems when they are required. Line 720 gives you the opportunity to trade your ore. The routine from line 7000 to 7150 works out the result of any attack by outsiders.

The clever little routine at the beginning of the game is provided by the routine from line 9000 to 9150 and is based on a routine developed by Matthew Boek. The routine takes a string 40 characters long, in this case line 9010, and then scrolls it from right to left across the screen. You can turn this routine to your own use by replacing the print statement in line 9010 with any 40 character string. The symbol used in the print statement is obtained from the "Q" key using SHIFT.


```

1 REM ASTEROID
2 REM "C" IS CLEAR SCREEN "D" IS CURSOR DOWN "E"
  IS REV ON "F" IS REV OFF
5 GOSUB 9000
10 DEF FNR(X)=INT(RND(1)*X)+1:GOSUB 3000
15 POKE 53280,6:POKE 53281,4
20 YR=YR+1:FK=FK+INT(FK/(2+FNR(18)))-FK/(FNR(15)+3))
  :GOTO 710
30 PRINT "COMPUTER REPORT:"
40 IF O<OJ*FK THEN 8000
50 IF F<FR*FK THEN 8010
60 IF CAC1 THEN 8020
70 IF FK<2 THEN 8030
80 IF FK<13 THEN PRINT "WARNING: POPULATION IS
  NEARING CRITICAL":GOSUB 8700
83 REM BLUE
90 IF O<2*OJ*FK THEN PRINT "WARNING: OXYGEN
  SUPPLY LOW":GOSUB 8700:REM BLUE
100 IF F<2*FR*FK THEN PRINT "WARNING: FOOD
  STOCKS LOW":GOSUB 8700:REM BLUE
110 IF CAC<2000 THEN PRINT "WARNING: CASH ON HAND
  RUNNING LOW":GOSUB 8700
112 REM BLUE
120 PRINT "THERE ARE";FK;"PEOPLE ON THE ASTEROID"
  :REM RED
125 PRINT "IN THE YEAR";YR
130 PRINT:PRINT "CASH: $";CA:PRINT "MAINTENANCE
  COSTS: $";RE
140 PRINT "OXYGEN COSTS: $";OC;"PER UNIT"
150 PRINT "OXYGEN NEED PER PERSON:"OJ:PRINT
  "FOOD:";F;"UNITS".
160 PRINT "PRESS ANY KEY TO CONTINUE":REM
  BLACK RED
170 GET Q$:IF Q$="" THEN GOTO 170
180 RETURN
700 GOSUB 30
710 PRINT "C"

```

```

720 PRINT "HOW MANY TONS OF ORE WILL YOU TRADE?"
    :REM RED
730 PRINT "OXYGEN USAGE WILL BE";AC;"PER TON
735 PRINT "EACH TON SELLS FOR";AP;:INPUT B
740 IF B*AC>0 THEN PRINT "NOT ENOUGH OXYGEN":FOR
    DD=1 TO 1500:NEXT:GOSUB 8700
745 IF B*AC>0 THEN GOTO 720
750 CA=CA+B*AP:O=O-B*AC:PRINT "I":GOSUB 30
760 PRINT "FOOD COSTS $";FC;"PER UNIT"
765 PRINT "EACH PERSON NEEDS";FR;"UNITS ($)";
    FC*FR;"EACH)"
770 PRINT "X";FK*FC*FR;"FOR THE ASTEROID"
780 PRINT "THIS WILL LAST";INT(F/(FR*FK));"YEARS
    AT THE"
785 PRINT "PRESENT POPULATION"
790 INPUT "HOW MUCH FOOD WILL YOU BUY";C
795 IF C*FC>CA THEN PRINT "NOT ENOUGH MONEY":
    GOTO 790
800 F=F+C*FC:CA=CA-C*FC:PRINT "I":GOSUB 30
850 PRINT "HOW MUCH OXYGEN WILL YOU BUY"
855 PRINT "PRESENT STOCKS WILL LAST FOR";INT(O/
    (OJ*FK))"YEARS"
860 PRINT "AT THE PRESENT LEVEL OF POPULATION";
    :INPUT D
865 IF D*OC>CA THEN FL=1
870 IF D*OC>CA THEN PRINT "NOT ENOUGH MONEY":FOR
    DD=1 TO 1500:NEXT:GOTO 850
880 PRINT "I":IF FNR(5)=2 THEN GOSUB 7000
2000 F=F-FK*FR:CA=CA-RE-D*OC:O=O+D-FK*OJ:GOTO 20
3000 Y=FNR(5):A$="THE ASTEROID IS DEAD":FK=FNR
    (40)+80
3005 CA=INT(7*(FNR(800)+700)/FNR(3))
3010 FC=FNR(7):AC=FNR(3)+1:F=FNR(500)+2000:G=2000
    +FNR(1500):AP=30*FNR(AC)
3020 RE=FNR(400)+200:FR=FNR(5)+1:OJ=FNR(3)+2:OC=
    FNR(7):O=2000-FNR(1500)
3030 RETURN

```

```

7000 J=FNR(6):PRINT "THE STATION HAS BEEN
      ATTACKED BY"
7010 ON J GOTO 7020,7030,7040,7050,7060,7070
7020 PRINT "A FLEET OF SIRIAN SHIPS":GOTO 7080
7030 PRINT "A RENEGADE SPACE MINERS":GOTO 7080
7040 PRINT "A MARTIAN SPACE PIRATES":GOTO 7080
7050 PRINT "A CALLISTAN CLAIM JUMPERS":GOTO 7080
7060 PRINT "A ROBOT CONTROLLED SURVEY SHIP":GOTO
      7080
7070 PRINT "A VENUSIAN BATTLE CRUISER"
7080 FOR DD=1 TO 1500:NEXT DD
7083 Z=1+INT(FK/(FNR(15)+1)):PRINT "THERE WERE";
      Z;"PEOPLE KILLED"
7085 GOSUB 8600
7090 Z1=FNR(250)+250:PRINT "DAMAGE WAS $";Z1
7100 Z2=FNR(300):Z3=FNR(300):PRINT "FOOD STOCKS
      HAVE FALLEN BY";Z2
7110 F=F-Z2:FK=FK-Z:CA=CA-Z1
7120 PRINT "Z3;UNITS OF OXYGEN HAVE BEEN LOST"
      :Q=Q-Z3
7130 PRINT "PRESS ANY KEY TO CONTINUE"
7140 GET P$:IF P$="" THEN GOTO 7140
7150 RETURN
8000 PRINT A$:PRINT "YOU RAN OUT OF OXYGEN IN
      YEAR";YR:GOTO 8500
8010 PRINT A$:PRINT "YOU RAN OUT OF FOOD IN YEAR"
      :YR:GOTO 8500
8020 PRINT A$:PRINT "YOU WENT BANKRUPT IN YEAR"
      :YR:GOTO 8500
8030 PRINT "YOUR POPULATION HAS FALLEN TO";FNR
      (26):CA=FNR(300)
8040 PRINT "DO YOU WISH TO GIVE UP NOW (1)"
8050 PRINT "OR DO YOU WISH TO ATTEMPT TO TRADE"
8060 PRINT "OUT OF TROUBLE (2)"
8070 INPUT B:IF B=1 THEN 8500      E
8080 PRINT "I HOPE YOU HAVE MADE THE RIGHT
      DECISION":GOTO 20

```

```

8500 FOR DD=1 TO 3000:NEXT:PRINT "I"
8510 POKE 53280,0:POKE 53281,11
8520 PRINT "XXXXXXXXXX" TAB(15) "FAREWELL":REM BLACK
8530 POKE 54296,15
8540 POKE 54277,17:POKE 54278,130
8550 POKE 54276,33
8560 POKE 54273,1:POKE 54272,205
8565 FOR DD=1 TO 500:NEXT DD
8570 POKE 54276,0:POKE 54277,0:POKE 54278,0
8575 FOR DD=1 TO 20:NEXT DD
8580 T=T+1:IF T<6 THEN GOTO 8540
8590 T=0:GOTO 8590
8600 POKE 54296,15
8610 POKE 54277,24:POKE 54278,136
8620 POKE 54276,65:POKE 54275,4:POKE 54274,200
8630 POKE 54273,22:POKE 54272,227
8640 FOR DD=1 TO 250:NEXT DD
8650 POKE 54276,0:POKE 54277,0:POKE 54278,0
8660 RETURN
8700 POKE 54296,15
8710 POKE 54277,17:POKE 54278,17
8720 POKE 54276,33
8730 POKE 54273,19:POKE 54272,63
8740 FOR DD=1 TO 150:NEXT DD
8750 POKE 54276,0:POKE 54277,0:POKE 54278,0
8760 FOR DD=1 TO 5:NEXT DD
8770 T=T+1:IF T<6 THEN GOTO 8710
8780 T=0:RETURN
9000 PRINT "I":POKE 53280,0:POKE 53281,2
9005 PRINT "XXXXXXXXXX":FOR DD=1 TO 40:PRINT CHR$
    (113):NEXT
9010 A$=" ASTEROID XXXXXXXXXXXX ASTEROID XXXXXXXXXXXX"
    :REM BLACK
9020 L=40
9030 FOR T=1 TO L
9040 PRINT "XXXXXXXXXX"
9050 IF (L-T)<1 THEN 9090
9060 PRINT TAB(L-T)LEFT$(A$,T);
9070 FOR DD=1 TO 100:NEXT
9080 NEXT

```

```
9090 FOR T=1 TO 39
9100 PRINT "XXXXXXXXXXXX"
9110 PRINT MID$(A$,T,L-T);
9120 FOR DD=1 TO 100:NEXT
9130 NEXT
9150 RETURN
```

GOLF 64 STYLE

Now here is your chance to try your golfing skills on the Commodore 64's computerised golf course. This is the perfect way to play golf, no wind or rain to spoil the play.

It is a game which will require you to choose the right club (to hit the ball with not to join!), then hit the ball straight and hard at the green. The object, of course, is to finish the game under par.

The computer will tell you the distance to the green at each hole and after each stroke will give you a tally of the total number of strokes you have taken, the number of strokes for the current hole and the par for the course.

When you get to within 20 yards (18 metres) of the hole you are considered to be on the green. The Commodore 64 will then putt out for you and tell you the result.

You have a choice of eight clubs, club number 8 giving you the greatest distance. Line 210 asks you which club you wish to use. Line 240 works out how far you have hit the ball.

Line 90 works out the distance from the tee to the cup for each hole and the routine from line 320 to 360 putts for you.

The hole course is controlled by the FOR/NEXT loop from line 70 to line 420.

The subroutine from line 500 provides the sound effects. Variable E is the strokes taken for each hole, variable T the total number of strokes, Z is the number of putts taken and Q is the par figure.

Golf 64 Style is another game which would be fun to experiment with. Using it as a basis for a Golf program, try adding new features, graphics for instance, drawing up a fairway showing where the ball has landed. You could also add a routine which would cause the ball to swing to the side. It would also be a good idea to add a more realistic putting feature.

```

10 REM GOLF 64 STYLE
20 POKE 53280,13:POKE 53281,5
30 REM "M" IS DOWN CURSOR
50 Q=0:T=0
70 FOR H=1 TO 18
80 E=0
90 D=INT(400*RND(1))+100
100 Q=Q+INT(D/70+.5)
110 PRINT "J":REM CLEAR SCREEN
120 PRINT TAB(10) "THIS IS HOLE:":H:REM RED
130 PRINT TAB(5) "NUMBER OF STROKES SO FAR:
    ";T:REM BLUE
140 PRINT TAB(8) "PAR AFTER THIS HOLE:":Q:REM
    GRAY 1
160 IF E>0 THEN PRINT TAB(10) "STROKES TAKEN:
    ";E:REM WHITE
170 PRINT TAB(12) "DISTANCE:":D:REM YELLOW
190 IF D=0 THEN GOTO 390
200 IF D<20 THEN GOTO 320
210 PRINT TAB(3) "WHICH CLUB WILL YOU USE (1-8)
    ";:REM BLACK
220 INPUT C
230 IF C<1 OR C>8 THEN GOTO 210
240 R=INT(C*20*(RND(1)+.5))
250 GOSUB 500
260 PRINT TAB(11) "YOU HIT":R;"YARDS":REM RED
280 D=ABS(D-R)
290 E=E+1
295 T=T+1
300 FOR DD=1 TO 3000:NEXT DD
310 GOTO 110
320 PRINT TAB(8) "YOU ARE ON THE GREEN":REM BLUE
340 FOR DD=1 TO 3000:NEXT DD
350 Z=INT(D/6)+1
360 PRINT TAB(10) "YOU TOOK":Z;"PUTTS":REM BLACK
380 E=E+Z
390 T=T+Z
400 PRINT TAB(9) "NUMBER OF STROKES":E:REM WHITE
410 FOR DD=1 TO 2500:NEXT DD
420 NEXT H

```

```

425 PRINT "I":POKE 53280,5:POKE 53281,13
430 PRINT TAB(10) "YOUR SCORE IS";T:REM
    BLACK
440 PRINT TAB(4) "THE PAR FOR THIS COURSE WAS";Q
450 PRINT TAB(3) "DO YOU WANT ANOTHER ROUND"
    (Y/N)";
460 INPUT A$
470 IF A$="N" THEN END
480 RUN
500 POKE 54296,15
510 POKE 54277,17:POKE 54278,17
520 POKE 54276,17
540 POKE 54273,12:POKE 54272,216
545 FOR DD=1 TO 50:NEXT DD
550 POKE 54277,0:POKE 54278,0:POKE 54276,0
560 RETURN

```


ELITE GRAMOPHONE COMPANY

Your computer has the ability to set up and manage a large number of variables. The uses of this amazing capability are only limited by your imagination. The "Elite Gramophone Company" demonstrates just how effectively your computer can handle simulated information, manipulating it according to ground rules laid down by the programmer.

This program allows you to experience the excitement and despair of running a factory in a harsh economic environment.

The computer will provide you with a weekly balance sheet. This will contain all the information you require, such as the current week of trading, capital on hand, stock on hand, the number of employees you have and their estimated productivity.

You must keep a close watch on the amount of capital you have on hand in order that you will be able to meet your production costs and weekly wages bill. This may be the key to your success or failure as a factory manager. The computer will give you the opportunity to hire and fire staff as the need arises, but you may find that the workers have a strong union if you try to put off too many of them.

You will be able to set weekly production targets (within the limits of your available capital and the productivity of your workers). Then you have the agonising wait for the sales figures.

At various times you will have to cope with union demands for higher wages, suppliers putting up the price of raw materials and the occasional disaster. There will be opportunities to raise the price of your gramophones but you must be careful. Don't price them right out of the market-place.

The object of the game is firstly to avoid the shame of bankruptcy and hopefully to go on to make a million dollars.

The program listing is quite long but playing it will reward the patience required to type it in.

The program consists of a control section, from lines 50 to 150, which calls up the relevant subroutines when they are required.

Line 30 calls up the subroutine which gives the title and sound at the start of the game. Line 40 calls up the routine beginning at line 1500. This subroutine sets up all the variables and parameters for the game.

Line 50 is the start of the game proper. This line increments the variable WE which counts the weeks.

GOSUB 950 in the next line provides the print out of the factory's status. This is called up at several points during the game to keep you up to date on what is happening.

GOSUB 1220 controls the hiring and firing of staff, plus any union reaction to your efforts to cut down on your work force.

GOSUB 1100 is your production line. It allows you to set a production target and then works out how many gramophones were actually produced.

GOSUB 790 is your sales team. This routine provides the good news or the bad news about the current week's sales. The sales figures can include gramophones from your warehouse as well as the week's production. Any unsold gramophones are automatically placed in storage.

GOSUB 360 controls the "unpredictables". These nasty little events include price rises for raw materials and union demands for higher pay. Not to mention the possibility of a warehouse fire.

```
10 REM ELITE GRAMOPHONE COMPANY
20 PRINT "J":REM CLEAR SCREEN "M" IS DOWN CURSOR
30 GOSUB 170
40 GOSUB 1500
50 WE=WE+1
60 GOSUB 950
70 GOSUB 1220
80 PRINT "J"
90 GOSUB 950
100 GOSUB 1100
110 GOSUB 950
```

```

120 GOSUB 790
130 GOSUB 360
140 CA=CA-INT(WG*WF)
150 GOTO 50
170 PRINT "J"
180 POKE 53280,8:POKE 53281,14
190 PRINT "XXXXXXXXXX"
200 PRINT "***** ELITE GRAMOPHONE COMPANY
*****J":REM RED RIGHTCRSR LT BLUE
210 FOR DD=1 TO 500:NEXT DD
220 POKE 53280,0:POKE 53281,3
230 GOSUB 2200
240 FOR DD=0 TO 15:POKE 53280,DD
250 FOR TT=1 TO 200:NEXT TT
260 NEXT DD
270 PRINT "J"
280 POKE 53280,6:POKE 53281,2
290 RETURN
360 REM UNPREDICTABLES
370 PRINT "J"
380 IF RND(1)>0.45 THEN GOTO 460
390 A=INT(10*RND(1))+1
395 GOSUB 2000
400 PRINT "XXXXXXXX" TAB(2) "UNIONS DEMAND A";A;"%
PAY INCREASE"
410 WG=INT(100*(WG+(A*WG/100)))/100
420 GOSUB 1620
430 PRINT "X" TAB(2) "PAY PER EMPLOYEE IS NOW $";WG
440 GOSUB 1620
450 PRINT "J"
460 IF RND(1)<0.80 GOTO 560
470 GOSUB 2000
475 PRINT "XXXXXXXX" TAB(4) "FIRE DESTROYS SOME OF
YOUR STOCK"
480 PRINT "X" TAB(6) "STANDBY FOR DAMAGE REPORT..."
490 GOSUB 1620
500 A=INT(SK/2*RND(1)*2)+1
505 IF A>SK THEN GOTO 500
510 SK=SK-A
520 PRINT "J":REM CLEAR SCREEN
525 PRINT "XXXXXXXX" TAB(4) "THERE WERE";A;"
GRAMOPHONES DESTROYED"

```

```

527 PRINT "0" TAB(4) "VALUE OF STOCK DESTROYED WAS
    $";A*SP
530 GOSUB 1620
540 PRINT "0" TAB(4) "STOCK ON HAND IS NOW";SK
550 GOSUB 1620
560 IF RND(1)>0.3 THEN GOTO 670
565 PRINT "3":GOSUB 2000
570 PRINT "00000" TAB(4) "SUPPLIER ANNOUNCES
    PRICE RISE"
585 GOSUB 1620
590 A=INT(100*CT/7*RND(1))/100
595 IF A<.01 THEN GOTO 590
610 PRINT "0" TAB(2) "THE COST OF MAKING
    GRAMOPHONES HAS"
620 PRINT "0" TAB(2) "GONE UP BY $";A;"EACH"
630 FOR ID=1 TO 1000:NEXT ID
640 CT=CT+A
650 PRINT "0" TAB(2) "IT NOW COSTS $";CT;"TO MAKE
    EACH ONE"
660 GOSUB 1620
670 IF RND(1)<0.65 AND MP<SP THEN RETURN
680 PRINT "3"
685 GOSUB 2000
690 PRINT "00000" TAB(2) "YOU HAVE AN OPPORTUNITY
    TO RAISE YOUR"
692 PRINT "0" TAB(16) "PRICE"
695 PRINT "0" TAB(2) "GRAMOPHONES NOW SELL FOR
    $";SP
700 GOSUB 1620
710 PRINT "0" TAB(2) "STATE THE PERCENTAGE OF
    YOUR PRICE"
715 PRINT "0" TAB(14) "INCREASE";
720 INPUT A
730 IF A>0 THEN Z=Z+A
740 SP=INT(100*(SP+A*SP/100))/100
750 GOSUB 1620
760 PRINT "3":PRINT "00000" TAB(4) "GRAMAPHONES
    NOW SELL FOR $";SP

```

```

770 GOSUB 1630
780 RETURN
790 REM SALE
800 PRINT "M" TAB(4) "STOCK ON HAND IS:-";SK
810 GOSUB 1620
820 PRINT "J"
825 GOSUB 2000
830 PRINT "MMMMM" TAB(8) "STANDBY FOR SALES REPORT"
840 GOSUB 1620
850 PRINT "J":FOR DD=1 TO 1000:NEXT DD
860 A=INT(RND(1)*SK/(Z/100))+1
870 IF A>SK THEN GOTO 860
880 GOSUB 2000
890 PRINT "MM" TAB(4) "TOTAL GRAMOPHONES SOLD..":
    FOR DD=1 TO 1000:NEXT:PRINT A
900 SK=SK-A
910 PRINT "MM" TAB(4) "INCOME FROM SALES: $";A*SP
920 CA=CA+INT(A*SP)
930 GOSUB 1620
940 RETURN
950 REM PRINTOUT
960 PRINT "J"
970 IF CA+SK<1 THEN GOTO 1370
980 IF CA+SK>999999 THEN GOTO 1640
985 GOSUB 2000
990 PRINT TAB(8) "JFACTORY REPORT: WEEK";WE:REM
    LIGHT BLUE
1000 PRINT "MOCAPITAL ON HAND IS $";CA
1010 PRINT "MOSTOCK ON HAND";SK;"GRAMOPHONES"
1015 PRINT "MNORTH $";SK*SP
1020 PRINT "MTHEY SELL FOR $";SP;"EACH"
1030 PRINT "MAND COST $";CT;"EACH TO MAKE"
1040 PRINT "MWORKFORCE IS";WF;"PEOPLE"
1050 PRINT "MTHEIR WAGES ARE $";WG;"EACH"
1060 PRINT "MTHIS WEEKS WAGES BILL IS $";WG*WF
1070 PRINT "MEACH PERSON CAN MAKE";PD;"GRAMOPHONES"
1080 PRINT "MA TOTAL OUTPUT OF";PD*WF;"GRAMOPHONES"
1090 RETURN
1100 PRINT "MHOW MANY GRAMOPHONES DO YOU WISH TO
    MAKE"
1105 INPUT MK
1110 IF MK=0 THEN RETURN

```

```

1120 IF MK*CT>CA THEN PRINT "NOT ENOUGH CAPITAL
      TO MAKE THAT MANY":GOTO 1100
1130 IF MK>PI*WF THEN PRINT "NOT ENOUGH WORKERS
      TO MAKE THAT MANY":GOTO 1100
1140 PRINT "J":PRINT "*****" TAB(3) "PRODUCTION
      TARGET FOR WEEK";WE;"IS";MK;
1150 MK=ABS(MK-(INT(RND(1)*2+1))*INT(RND(1)*10))
1160 GOSUB 1620
1170 PRINT:PRINT "X" TAB(6) "TOTAL MADE IN WEEK";
      WE;"WAS";MK
1180 SK=SK+MK
1190 CA=INT(CA-CT*MK)
1200 GOSUB 1620
1210 RETURN
1220 REM PEOPLE
1230 PRINT "HOW MANY PEOPLE DO YOU WISH TO HIRE";
1235 INPUT A
1240 WF=WF+A
1250 PRINT "J":PRINT "*****" TAB(6) "TOTAL
      WORKFORCE IS NOW";WF
1260 GOSUB 1630
1270 IF A<>0 THEN GOTO 1350
1280 PRINT "HOW MANY PEOPLE DO YOU WISH TO FIRE";
1285 INPUT A
1290 IF A=0 THEN GOTO 1350
1300 IF A>WF THEN GOTO 1280
1310 A=INT(RND(1)*A+1)
1320 GOSUB 1630
1330 PRINT "THE UNION WILL ONLY ALLOW YOU TO
      FIRE";A
1340 WF=WF-A
1350 GOSUB 1630
1360 RETURN
1370 REM BANKRUPTCY
1380 L=0
1385 GOSUB 2000
1390 PRINT "J":PRINT "*****":REM CLEAR SCREEN
      HOME DOWN CURSOR
1392 POKE 53280,2:POKE 53281,0

```

```

1395 PRINT TAB(10) "##### BANKRUPT #####":FOR
    DD=1 TO 750:NEXT DD:REM YELLOW
1400 L=L+1:IF L=10 THEN GOTO 1430
1405 POKE 53281,2:POKE 53280,0:FOR DD=1 TO 500:NEXT
1410 GOTO 1390
1430 PRINT "J":GOSUB 1620
1440 PRINT "X"YOU KEPT THE ELITE GRAMOPHONE
    COMPANY"
1445 PRINT "XRUNNING FOR";WE;"WEEKS":FOR DD=1 TO
    5000:NEXT DD
1450 PRINT "J":PRINT "X" TAB(4) "DO YOU WANT TO
    TRY AGAIN (Y/N)"
1460 GET A$:IF A$="" THEN 1460
1470 IF A$="Y" THEN RUN
1490 PRINT "J":PRINT "XXXXXXXXXX" TAB(16) "GOODBYE"
1495 GOTO 1495
1500 REM VARIABLES
1510 CA=500+INT(500*RND(1))+1
1520 SK=100+INT(500*RND(1))+1
1530 SP=10+INT(5*RND(1))+1
1540 CT=2+INT(5*RND(1))+1
1550 IF CT>SP THEN GOTO 1530
1560 WF=7+INT(10*RND(1))+1
1570 WG=12+INT(RND(1)*(SP*5))
1580 PD=5+INT(5*RND(1))+1
1590 WK=1
1600 Z=1
1610 RETURN
1620 FOR DD=1 TO 2500:NEXT DD:RETURN
1630 FOR DD=1 TO 1500:NEXT DD:RETURN
1640 C=0
1650 PRINT "J":PRINT "XXXXXXXXXX":REM CLEAR SCREEN
    HOME DOWN CURSOR
1655 POKE 53280,2:POKE 53281,4
1655 PRINT TAB(3) "##### YOU'VE MADE A MILLION
    #####":REM WHITE
1660 C=C+1
1670 GOSUB 2000:FOR DD=1 TO 200:NEXT DD
1680 PRINT "XXXXXXXXXX"
1690 PRINT TAB(3) "#####
    #####":FOR DD=1 TO 250:NEXT DD

```

```

1700 IF C=10 GOTO 1450
1710 GOTO 1650
2000 R1=INT(50*RND(1))+4
2010 R2=INT(100*RND(1))+100
2020 POKE 54296,15
2030 POKE 54277,64:POKE 54278,130
2040 POKE 54276,17
2050 POKE 54273,R1:POKE 54272,R2
2060 FOR DD=1 TO 250:NEXT DD
2070 POKE 54276,0:POKE 54277,0:POKE 54278,0
2080 RETURN
2200 POKE 54296,15:A=54277:S=54278:W=54276
2210 POKE A,130:POKE S,130
2220 POKE W,128
2230 HF=54273:LF=54272
2240 POKE HF,14:POKE LF,107
2250 FOR DD=1 TO 350:NEXT DD
2260 POKE W,0:POKE S,0:POKE A,0
2265 FOR DD=1 TO 20:NEXT DD
2270 POKE A,15:POKE S,20:POKE W,128
2280 POKE HF,12:POKE LF,216
2290 FOR DD=1 TO 75:NEXT DD
2300 POKE W,0:POKE A,0:POKE S,0
2305 T=T+1
2310 FOR DD=1 TO 200:NEXT DD
2320 IF T=4 THEN T=0:RETURN
2330 GOTO 2200

```

READY.

Excitement will grip you the moment you put the first of these games up on your screen. William Roberts has devised the programs in **Challenging Games for the Commodore 64** with a great sense of entertainment and stimulation.

Enemy-blasters, outwit-the-computers, shoot-outs—they're all here! So boot the system and start the fun.



Another great book from INTERFACE PUBLICATIONS

ISBN 0 907563 48 1